Iranian Journal of Numerical Analysis and Optimization
Vol. ??, No. ??, ??, pp ??
https://doi.org/??
https://ijnao.um.ac.ir/

Research Article

Convex-hull based two-phase algorithm to solve capacitated vehicle routing problem

M. Afsharirad*, and A. Hashemi Borzabadi

Abstract

The goal of this paper is to present a two-phase convex hull-based algorithm for the capacitated vehicle routing problem (CVRP), consisting of clustering and routing phases. First, a K-means-based algorithm is proposed for the clustering phase, where the centroids are updated according to the convex hull of the assigned points. Furthermore, a convex-hull-based algorithm is suggested for the routing phase, which iteratively inserts unrouted points into the convex hull. To improve the routes, an ant colony optimization algorithm is applied. It is shown that the proposed method has a time complexity of order $o(n^2 \log n)$, where n is the number of customers. For performance evaluation, we utilize CVRP benchmark samples and compare

*Corresponding author

Received ??? ; revised ??? ; accepted ???

Maria Afsharirad
Department of Applied Mathematics, University of Science and Technology of Mazandaran, Behshahr, P.O. Box: 48518-78195, Mazandaran, Iran. e-mail: maria.afsharirad@gmail.com

Akbar Hashemi Borzabadi
Department of Applied Mathematics, University of Science and Technology of Mazandaran, Behshahr, P.O. Box: 48518-78195, Mazandaran, Iran. e-mail: akbar.h.borzabadi@gmail.com

How to cite this article
Afsharirad, M. and Hashemi Borzabadi, A. , Convex-hull based two-phase algorithm to solve capacitated vehicle routing problem. *Iran. J. Numer. Anal. Optim.*, ??; ??(??): ??-??. ??

the results to those of other two-phase CVRP algorithms. The proposed clustering method combined with common routing techniques, as well as the K-means clustering method paired with the proposed routing approach, yields highly favorable results in some instances. Moreover, the proposed two-phase method outperforms other approaches in certain instances.

AMS subject classifications (2020): Primary 90C27; Secondary , 90C59.

Keywords: Capacitated vehicle routing problem, K-means algorithm, Convex hull, Ant colony optimization

1 Introduction

The vehicle routing problem (VRP) is one of the most well-known problems in combinatorial optimization due to its wide applicability in fields such as public transportation, waste collection, and drone routing. In its typical form, the VRP involves finding routes for a fleet of vehicles with limited capacity to serve customers. These routes start from a central node called the “depot”, return to it after visiting customers (within vehicle capacity constraints), and aim to optimize an objective function—commonly minimizing total distance or total service time.

CVRP might be divided into two phases, the first phase is “clustering”, in which customers are assigned to vehicles. The second phase is “routing”, which determines the optimal route of all vehicles in their cluster. It is clear that the second phase is the well-known traveling salesman problem (TSP). Since CVRP contains TSP and Bin packing problem as its special case, therefore it is classified as an NP-hard problem.

A two-phase convex-hull based (CHB) heuristic method is provided in this paper. First, customers are clustered by a convex-hull based K-means (CH-means) algorithm. Since the length of the whole TSP tour should be minimized in CVRP, in CH-means algorithm, the centers are updated as the mean point of the convex hull of the points assigned to each cluster. Accordingly, any cluster contain all points on the line segment joining any two points in it. However adding points in any step is the same with the

K-means algorithm and is based on the minimum distance from the center, but since closest points to the center are located inside the respective convex hull, they will be assigned to that cluster.

Furthermore, the fitness of any step's solution is calculated by a proposed routing algorithm, and it is saved. At the end of the clustering phase, the best found solution is chosen for improvement, which is not necessarily the last solution found. This increases the computational complexity of the algorithm, but it allows the selection of the best clustering.

The routes are built by the proposed convex-hull based routing algorithm (CH-Insertion). Our idea for routing is to construct the convex hull of the points and to insert unassigned points by breaking an edge of the polygon into two edges. This is based on the well known property for Euclidean TSP: The order in which the points appear in an optimal TSP tour must be the same as the order in which these points appear on the convex hull, [30]. Finally, the routes are improved with a meta-heuristic algorithm. Ant colony optimization algorithm has been chosen, due to its satisfactory performance on TSP.

The paper is organized as follows: The literature is reviewed in Section 2 in three categories: Variations, applications and approaches. The problem is defined in Section 3. Section 4 is devoted to the two-phase CHB heuristic. In its first subsection, the CH-means algorithm is presented, and the second subsection explains the CH-insertion algorithm. The last subsection discusses ant colony optimization algorithm. Finally, Section 6 presents numerical results and concludes the paper.

2 Related works

The problem was first introduced by Dantzig and Ramser [15]. They applied the problem to petrol deliveries and proposed the first approximation algorithm based on matching. In the following, we review related literature in three categories. First, we state variations of the problem. Then we list the most major applications of VRP. Finally, we discuss different approaches of the problem in literature.

2.1 Variations of VRP

Numerous variations have been provided by researchers for VRP due to its variety of applications. Some famous variations are as follows:

Capacitated vehicle routing problem (CVRP) is the most closely related version to the original VRP in which a positive number is assigned to each customer as its capacity or quantity of its demand, see [57].

The VRP with time windows (VRPTW) is a variation of CVRP in which serving any customer must be done within a determined time interval, [52, 53]. There is also another closely related variation of VRP, dealing with online real-time demands. A hybrid meta-heuristic algorithm based on genetic algorithm and tabu search for on-line VRP, is provided in [31].

The VRP with profits (VRPP) is a maximization problem, where all customers do not have to be serviced. The problem is to visit all customers at most once in order to maximize the sum of collected profits according to a vehicle time limitation, [25].

Open VRP (OVRP) is another variation of the problem in which vehicles do not have to return to the depot at the end of their route, [48].

Multi-depot VRP (MDVRP) has multiple nodes as the depot, and the problem is also to assign each vehicle to each depot, [34]. The classic form of the MDVRP in which all vehicles start and end their route at the same depot, is considered in [46].

The VRP with Drones (VRPD) is an extension of CVRP, where not only trucks but also drones are used to service customers. One distinctive feature of the VRPD is that a drone may travel with a truck, take off from its stop to serve customers, and land at a service hub to travel with another truck as long as the flying range and loading capacity limitations are satisfied, [19, 59]

For comprehensive reviews of VRP refer to [5, 4, 35].

2.2 Applications

Recent applications of the VRP have expanded into specialized domains. Authors in [16] provided a comprehensive review of the police patrol routing problem, highlighting how VRP models ensure balanced patrol allocation while minimizing response times and workload disparities across patrol zones. Their work underscores VRP's flexibility in adapting to public-safety contexts, where route equity and real-time adjustments are vital in urban environments. A multi-objective VRP is applied in [36] to optimize real-world postal delivery at scale. They balance delivery efficiency and service fairness—incorporating objectives such as minimizing total distance and regulating driver workload. Through meta-heuristics, they demonstrate substantial cost savings while maintaining operational equity across a large delivery network.

Unmanned aerial vehicle (UAV) applications are reviewed in [56], in VRP contexts—emphasizing disaster relief, surveillance, and agricultural logistics. Their meta-analysis captures how drone-based VRPs address reach limitations of ground fleets, introducing constraints like battery life and communication reliability. This study solidified VRP's extension into UAV-coordinated systems.

Building on these foundations, recent work has advanced VRP in multi-modal and uncertain environments. VRPD-DT is introduced in [27], a vehicle-and-drone routing framework that integrates dynamic traffic prediction using machine learning. Their real-time VND heuristic outperformed static models, showing improved delivery times under fluctuating conditions. On the same front, authors in [14] survey truck-drone cooperative VRPs, categorizing operational modes—from synchronous to independent operations—and summarizing over 200 studies with implications for last-mile delivery, reconnaissance, and patrol. A PRISMA-based review of satellite depots in urban logistics is performed in [54], finding that roughly half of VRP designs incorporate cross-docking via intermediate warehouses. Their review also highlights significant gaps in stochastic and dynamic VRP modeling.

Together, these studies illustrate VRP's evolution beyond classical delivery models toward multi-objective, multi-modal, and dynamic frameworks. Integrating drones (see [27, 14]) and satellite depots (see [54]) supports granular, responsive logistics systems. In public-safety and postal services, VRP's ability to balance equity and efficiency remains critical, as demonstrated in [16, 36]. The confluence of these advancements reflects VRP's growing relevance in addressing complex, real-world routing challenges that demand adaptability, real-time response capabilities, and multi-objective optimization.

2.3 Approaches

VRP is the process of selecting feasible routes out of exponentially many selections of any combination of customers with determined demands. There exist three types of integer programming formulations in literature for VRP, which are based on: Commodity flow formulations [10], vehicle flow models [29], and set partition problem [1]. According to its NP-hardness, exact methods are suitable for small instances only. Branch and bound, branch and cut and dynamic programming algorithms are exact methods applied by researchers. A comprehensive overview of exact methods for CVRP and its other variations, is provided in [58].

The first algorithmic approach for VRP, has been provided in [15]. Their algorithm was a simple matching-based heuristic. Afterwards, the Dantzig and Ramser's approach was improved by an effective greedy approach called the savings algorithm, [12]. Generally heuristics for VRP are clustered into two categories:

1. Constructive methods: In this type of methods, tours are constructed gradually by adding nodes to them or by combining subtours in a way not to exceed the capacity. Savings algorithm in [12] is the most famous heuristic of this type.

2. Two-phase methods: These methods solve the problem in two phases, the clustering phase and the routing phase. According to [50], there are two types of two-phase methods, the cluster-first, route-second and the route-first, cluster-second.

This paper suggests a second type heuristic in "cluster-first, route-second" order. So we focus on the same two-phase algorithms in literature. Clustering phase might be done by different approaches. A sweep algorithm is proposed in [22]. The authors consider the depot as the origin of the plane and order customer points according to their argument in polar coordination system. Then the points are assigned respectively to vehicles up to fulfilling its capacity. In the second phase, they propose an iterative procedure to improve the route of any vehicle. Sweep algorithm has been widely used for the first phase of the algorithm.

A popular two-phase algorithm is provided in [21]. The authors apply the generalized assignment problem for clustering phase and find routes of any cluster, using any TSP method. Sweep algorithm for clustering phase and nearest neighbor approach for routing phase for public transportation problem is applied in [42], in which capacity of the vehicle may vary during their tour, since some passengers may end their trip before the route is completed. The problem of routing drones, in which the combination of sweep algorithm and genetic algorithm is applied to solve VRPD, [19]. The original sweep algorithm in [22] for VRPTW, is applied in [26]. Authors in [17] implemented first-stage customer clustering, then second-stage routing subproblems separately for conventional and electric fleets, enhancing mixed-fleet planning. A two-stage approach is suggested in [32]: first, reduce the network using A* shortest paths; second, route using an enhanced GA with large-neighborhood search for vehicles with charging considerations. Moreover, a novel 2-phase approach is suggested in [44], which strategically separates customer location/routing decisions and operational routing, integrating a hybrid MILP and MCDM approach for service-option VRPs.

Other commonly used methods for clustering are based on data-mining algorithms. The K-means algorithm is a common procedure for the clustering phase of CVRP. The goal is to divide data into K clusters, to minimize the inner-group dissimilarity. Cluster's dissimilarity is measured by the average distance between cluster center and dataset points.

The multi-depot heterogeneous fleet VRPTW is provided in [18]. Authors provide a 3-phase hierarchical procedure. In clustering as the first phase, a heuristic is integrated into an optimization framework. Clusters of nodes are defined first, then points of the clusters are sequenced on the related tours and finally, the routing and scheduling for each tour are separately found, in terms of the original nodes. A multi-phase algorithm based on K-means clustering is developed in [33] for multi-depot VRP. The savings algorithm for the cumulative VRP with limited duration is developed in [11] where the load is also considered in the objective function as well as distance. The authors provide a K-means algorithm for clustering in which centroids are updated iteratively, according to a square error function, based on the angle of the line from the depot and any point of the cluster. The K-means method is used for clustering in [38], in order to adjust size of the clusters, decide whether to exchange points between two clusters or not, by calculate the value of the objective function. The authors in [13] used three hierarchical algorithms: K-means, K-medoids and DBScan. Generally, their clustering method is to randomly determine the first K centers and then assign each customer point to the closest center. New center of any cluster switch to the point possessing the mean value of all objects in that cluster. The procedure repeats till center points remain unchanged. Since clusters may not be feasible at the end, capacity control of the clusters is done by an MILP to make them ready for the routing phase.

Authors in [51] initialized a number of clusters and centroids manually and assign any customer to the closest center and in any iteration update centers as the mean value of the cluster. They also apply saving matrix method in the routing phase. K-medoids clustering is evaluated for CVRP in [6]. An existing meta-heuristic is applied for routing of each generated cluster. Recently, the bi-objective green delivery and pick-up problem has been considered in [20]. The authors used a K-means based algorithm for clustering and a genetic algorithm, for routing phase.

An overview of papers targeting different variations of VRP with two-phase methods is given in Table 1.

2.3.1 CHB approaches for routing phase

Since the order of nodes in the optimal tour of TSP with Euclidean distance, is the same order in their convex-hull, there are plenty of algorithms in the routing phase of VRP, based on convex-hull of the points. Accordingly, a fast algorithm is proposed in [23], in which the convex hull of the points are constructed and then for each point not contained in the convex-hull find the edge of it, such that the saving introduced in [12] from the non-contained

Table 1: An overview of cluster-first, route-second two-phase methods for VRP

Authors	**The first phase approach**	**The second phase approach**	**Problem**
Gillett and Miller (1974), [22]	Sweep algorithm	An iterative procedure to improve routes	VRP
Fisher and Jaikumar (1981), [21]	Generalized Assignment Problem(GAP)	Any TSP method	CVRP
Nurkahyo et al. (2002), [42]	Sweep algorithm	Nearest neighbor search	VRP for public transport
Dondo and Cerda (2007), [18]	Hierarchical procedure	Optimization framework	Multi-depot VRPTW
Nallusamy et al. (2009), [38]	K-Means	Genetic algorithm	Multi-VRP
Luo and Chen (2014), [33]	K-means	3-phase algorithm: Local search +Binary tournament +Cluster adjustment	Multi-depot VRP
Cinar, Gakis, and Pardalos (2016), [11]	K-means	Modified Savings Matrix	Cumulative VRP
Comert et al. (2017), [13]	K-means, K-medoids and DBScan	MIP	VRPTW
Singanamala, Reddy, and Venkataramaiah (2018), [51]	K-means, Savings matrix method	Ant colony	Multi-depot VRP
Bruwer (2018), [6]	K-medoids	Ruin and Recreate method	CVRP
Hertrich, Hungerländer, and Truden (2019), [26]	Sweep algorithm	The same order in sweep algorithm	VRPTW
Euchi and Saduk (2020), [19]	Sweep algorithm	Genetic algorithm	VRPD
Chen, Gu, and Gao (2020), [9]	Exact assignment algorithm	Genetic algorithm	Multi-depot VRPTW
Fatemi-Anaraki et al. (2021), [20]	K-means	Genetic algorithm	Delivery and pick-up problem
Ding, Li, and Hao (2023), [17]	Clustering algorithm (Nearest ID)	Heuristic routing	VRP with mixed fleet
Liu et al. (2023), [32]	Improved sweep algorithm	The same order in sweep	2-depot CVRP
Pourmohammadreza and Jokar (2023), [44]	A-star search algorithm	enhanced genetic algorithm	VRP with charging relief
Sehta and Thakar (2023), [49]	sweep algorithm	Genetic algorithm	CVRP

node is minimal. In this way, all non-contained nodes are assigned to an edge of the convex-hull, and among them, a node with the minimum saving number is added to the convex-hull. The procedure is repeated until all nodes are covered. They also showed that the computational complexity of the algorithm is of the order $o(n^2 \log n)$, while its worst case is unknown. The algorithm has been modified in [55] to improve the results.

Authors in [28] proposed a strange heuristic for TSP based on convex hull. A blob is located over the set of nodes which are projected into the lattice. The blob is gradually reduced until it passes all nodes in its edge. The initial shape is the convex hull of the points. It is then shrunk by systematically removing some of its constituent particle components. The points act as attractants to the material, effectively "snagging" the material at the locations of uncovered nodes and affecting its subsequent morphological adaptation. As the material continues to shrink all data points, it is becoming a concave area covering all nodes. The classic TSP solutions are enhanced with a convex hull insertion method, in [24], by providing a systematic and fast way to construct near-optimal tours. Authors in [41] enhanced classic TSP solutions with a convex hull insertion method, providing a systematic and fast way to construct near-optimal tours.

A CHB method has also been applied for VRP. The mathematical model for CVRP is considered in [45]. Authors provide a decomposition algorithm for capacity constraints and apply a separation problem to identify nodes violating this constraint. They use convex hull of incidence vectors of all TSP tours, and these tours are tested to find the violated capacity constraints. The convex hull of the points is used in [43] to measure the visual attractiveness of the solution. A saving based algorithm is applied to solve an extended version of VRP. The min-max multi VRP is introduced in [39], in which there are multiple depots, and the objective is to minimize the maximum length of the tour traversed by vehicles. The author uses the convex hull of all nodes containing customer and depot points to find the whole region at hand and then applies Carlsson algorithm in [8] to partition the region for multiple depots.

Authors in [47] consider four criteria to measure visual attractiveness of the routes containing "number of nodes belong to more than one convex hull".

They propose a heuristic in which the farthest node from the depot is chosen as the seed of a new route, then the surrounding nodes are added to it, until the capacity limit is reached. The whole procedure is repeated until all nodes are routed. After building routes, they find the nodes locating in the convex hull of another route and apply a “Merge-And-Rebuild” process to fix it. In a recent research, the convex hull of customer locations is applied to select initial seed clients, followed by an exchange operator to improve solutions in VRPTW, [49].

3 Problem statement

The problem addressed in this paper consists of designing efficient routes for K identical vehicles to service a set of customers with known demands. More precisely, CVRP is described as an undirected weighted graph $G = (V, A, c)$, where $V = \{0, 1, \ldots, n\}$ is the set of vertices, in which point 0 is the depot point and $\{1, \ldots, n\}$ is the set of customers, and $A = \{(i, j) :) \in V \times V,\ i \neq j\}$ shows the set of arcs. homogeneous vehicle fleets, each with capacity Q, start their route from the depot and end to it after visiting a subset of customers according to their limited capacity. Moreover, d_{ij} is the Euclidean distance between nodes i and j, while $q_i > 0$ shows the customer i's demand. The problem is solved under the following constraints:

1. Each customer is serviced only once by one vehicle.
2. Each vehicle must start and end its route at the depot.
3. Total demand met by each vehicle cannot exceed Q.

4 The two-phase CHB heuristic

CVRP is solved using a proposed two-phase CHB heuristic.

Clustering Phase: This phase is to assign customers to clusters.

- Customers are partitioned into K clusters using a novel CH-means algorithm, where each cluster is assigned to a vehicle.

- Cluster centroids are updated based on the convex hull of the nodes within each cluster.

Routing Phase: This phase is to find efficient routes between nodes of any cluster.

- Efficient routes between nodes in each cluster are constructed using the proposed CH-insertion heuristic.
- The routing procedure is executed at the end of any iteration of the clustering phase to retain the best solution by the end of Phase 1.
- The fitness of a solution is defined as the total length of all routes in that solution.
- Finally, an ant colony optimization procedure refines the routes for further improvement.

Algorithm 1 explains the CHB heuristic.

4.1 Clustering phase: CH-means method

A CH-means algorithm is provided here, while the number of clusters is already known, and it is equal to the number of vehicles. Centroids are updated after a complete iteration of the algorithm. The procedure is repeated until the distance between centroids in two consecutive iterations in all clusters, does not exceed a predetermined threshold. The steps of the CH-means algorithm are described in the following.

Step 1: Create initial centroids and distance matrix

Initially, the whole region is divided into K regions by plotting K lines originating from depot. To do this, customer points are sorted according to their arguments in polar coordination, where the depot is assumed as the origin. The whole region containing customers is identified by polar coordination $[\theta_{\min}, \theta_{\max}]$. The region is divided to K cones by $K-1$ equidistant rays

Algorithm 1 Two-phase CHB algorithm for CVRP

Input: $V = \{0\} \cup \{1, \ldots, n\}$ depot point and customer points, q demand vector, Q vehicle capacity, $[X, Y]$ Cartesian coordination of vertices and K: No. of vehicles, parameter ϵ.

Output: K routes start from and end to depot point $0 \in V$, s.t. all customer points exists in exactly one route and the sum of customer demands in all routes do not exceed Q.

while The distance between centroids is more than ϵ **do**
  Cluster customer points into K clusters $clus_j$, $j = 1, \ldots, K$, by Algorithm 2.
  for $j = 1, \ldots, K$ **do**
    Find initial route of $Clus = clus_j \cup \{0\}$ by Algorithm 3, call it $tour_j$.
  end for
  Calculate the fitness of $Clus$. Save it if it is the best clustering found by now.
end while
Find the best clustering and its routing and call it $tour^*$.
Improve $tour^*$ by the ant colony optimization of Algorithm 4.

originating from depot. The middle points of cones $\frac{r}{2}$ far from the depot, are forming initial centroids, where r is the half of the maximum distance from depot. Figure 1 shows two different examples.

Distance matrix D is calculated with entities d_{ij} as the Euclidean distance between customer point i and centroid j.

Step 2: Assign customers

The first assignment is related to the minimum entity of D, say d_{pq}, if vehicle q has enough empty capacity for customer p. This step is repeated till all points are assigned.

Remark 1. In some cases, some points might be remained unassigned, while no other cluster has enough empty capacity to meet their demand. There are

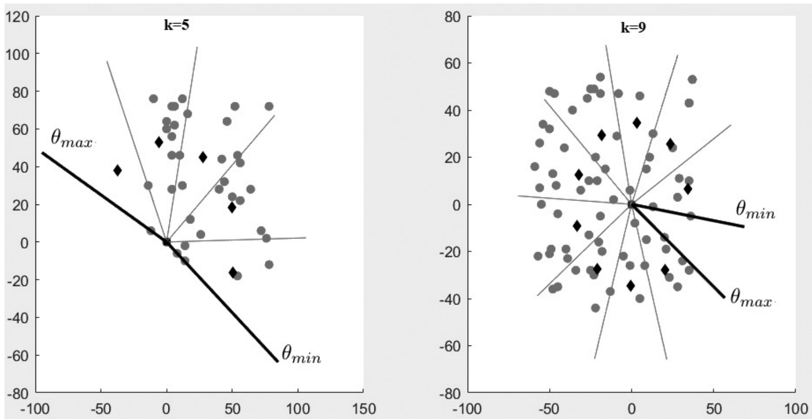

Figure 1: Dividing the whole region by K rays, diamond black points are initial centroids.

few approaches in literature proposing some methods to convert uncapacitated clusters to capacitated ones, [2, 7]. Most researches open a new cluster for remained points, [40, 37]. Since with the proposed method, this case was so rare, a new cluster is opened for these points instead of undertaking the cost of cluster improving methods. However, this kind of samples will be pointed out in Section 6.

Step 3: Calculate fitness

At the end of step 2, the fitness of the obtained clustering is calculated. This is done by creating a route for any cluster according to the proposed procedure in subsection 4.2. The fitness is saved and at the end of the whole algorithm, the best solution is selected to be improved by an ant colony optimization method.

Step 4: Update centroids and distance matrix

Centroids are updated, after a complete assignment. The centroid of any cluster is the mean point of the vertices of the convex hull of all points assigned to that cluster. Subsequently, distance matrix D is updated according to new centroids. Steps 2 and 3 are repeated until the distance between all corresponding centroids in two consecutive iterations do not exceed a predetermined threshold. This is mentioned by *approximately unchanged centroids* in Algorithm 2. This algorithm explains the whole procedure in CH-means algorithm.

4.2 Routing phase: CH-insertion method

This section presents the CH-insertion algorithm for routing points within a cluster, which is clearly a TSP. Firstly, the convex hull of the current cluster's points is computed to form the initial polygonal route. Afterwards, for each unrouted point on that cluster say p, all polygon edges say (i, j) are identified, for which the triangle $i - p - j$ is acute. Next, any unrouted point is assigned to its closest edge having this property. The insertion cost $height_p$ is calculated as the perpendicular height from p to edge (i, j), guaranteed to lie within the triangle due to the acute angle condition. Among all unrouted points p, the one, possessing minimum $height_p$ is found and its associated edge (i, j) is broken into two sides (i, p) and (p, j). The procedure is repeated until all nodes are incorporated into the route. The complete algorithm is formalized in Algorithm 3.

Figure 2 shows steps of Algorithm 3 for an instance. Part (a) is the initial step in which the convex hull of the points is found. In the next step, distance of any point in $\bar{N}$ to its associated side is calculated, and the nearest one (E) is selected and is added to the polygon. Point B is also added in the next step. Part (f) shows the final rout. It should be noted that the clusters are feasible at the beginning of the CH-insertion algorithm, since these points are resulted from the previous clustering step.

Algorithm 2 CH-means algorithm for clustering

Input: $V = \{0\} \cup \{1, \dots, n\}$ depot point and customer pints, q demand vector, Q vehicle capacity, $[X, Y]$ Cartesian coordination of vertices, K number of vehicles.

Output: Clusters $clust_j$ for $j = 1, \dots, K$.

Set initial centroids as the middle point of cones $\frac{r}{2}$ far from the depot. ▷ r: half of the maximum distance from depot

Construct distance matrix D. Let $best = \infty$ ▷ (Initialization)

Let $cap_j = 0$ for $j = 1, \dots, K$. ▷ (cap_j is the occupied amount of capacity of vehicle K).

while All centroids approximately remain unchanged **do**

  let $AC = \phi$. ▷ (AC is the set of assigned customers).

  while $AC \neq \{1, \dots, n\}$ **do**

    $d_{rp} = min\{d_{ij}, i = 1, \dots, n,\ j = 1, \dots, K\}$.

    if $d_{rp} < \infty$ **then**

      if $cap_p + q_r \leq Q$ **then**

        assign customer r to vehicle p: $clust_p = clust_p \cup \{r\}$,

        $cap_p \leftarrow cap_p + q_r$ ▷ (Update occupied capacity of vehicle p)

        Let $d_{rj} = \infty$, for $j = 1, \dots, K$ ▷ (Avoid reassigning customer r)

        Let $AC \leftarrow AC \cup \{r\}$,

      else

        Let $d_{rp} = \infty$ ▷ (Avoid reconsidering customer r for vehicle p)

      end if

    else

      Open a new cluster $clust_{K+1}$.

    end if

  end while

  Find the convex hull of all clusters.

  Update centroids of clusters as the mean value of vertices of their convex hull.

  Update distance matrix D, according to new centroids.

end while

Algorithm 3 CH-insertion algorithm for routing

Input: Clusters $clust_j$ for $j = 1, \dots, K$. $[X, Y]$: Cartesian coordination of vertices.

Output: Tours $tour_j$ for $j = 1, \dots, K$, each starts and end to depot.

for $j = 1, \dots, K$ **do**
    Let $H =$ the convex hull of points in $clust_j$,
    let N be the points in H, $\bar{N} = clust_j \setminus N$. ▷ (Initialize routed and unrouted points)
    while $\bar{N} \neq \phi$ **do**
        for $h \in \bar{N}$ **do**
            Let $(i,j)_h = arg \min_{(i,j) \text{ is a side of H}} \{height_h^{ij} \text{s.t. triangle i-h-j is acute}\}$.
        end for
        Let $height_p = min\{height_h^{(i,j)_h},\ h \in \bar{N}\}$.
        Update H by breaking the side $(i,j)_p$ into two sides (i,p) and (p,j).
        Update $N \leftarrow N \cup \{p\}$ and $\bar{N} = \bar{N} \setminus \{p\}$.
    end while
    $tour_j$ is the set of ordered points in polygon H starting from 0.
end for

4.3 Computational complexity of the two-phase CHB heuristic

This section shows that the complexity of the CHB algorithm is polynomial in terms of the number of customers.

Lemma 1. The computational complexity of the proposed two-phase CHB heuristic algorithm is $O(n \times IT \log(n))$, where IT is the maximum number of iterations and $IT \approx \gamma \frac{1}{\epsilon}$, with $\gamma \in [5, 10]$ depending on the benchmark samples used.

Proof. The computational complexity is analyzed in two phases:

Clustering Phase:

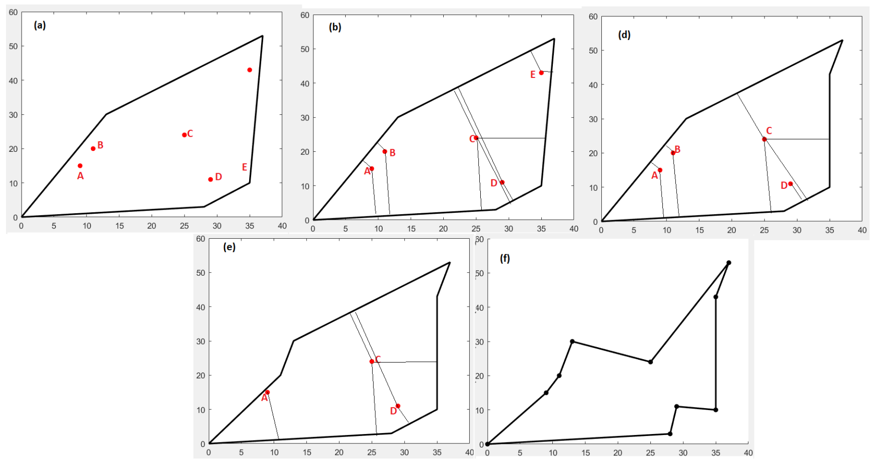

Figure 2: Steps of CH-insertion Algorithm 3

- Constructing the distance matrix for n nodes and K clusters costs $O(nK)$.
- Finding the minimum element in the distance matrix, which contains nK elements, requires $O(nK\log(nK))$.
- Assigning nodes to vehicles until all points are assigned takes $O(n)$.

Hence, the complexity of clustering in one iteration is $O(nK\log(nK))$, which simplifies to $O(n^2\log(n))$ when $K = O(n)$. At the end of any single iteration of clustering, routing procedure is implemented. First, we compute the number of computations in one single cluster.

Routing Phase:

- In one cluster, each vehicle serves approximately $h = \frac{n}{K}$ customers.
- Constructing the convex hull of h points using standard algorithms (e.g., CGAL or SciPy) costs $O(h\log(h))^{\star}$.
- Calculating the distances of all inner points from all sides of the convex polygon takes $O(h^2)$.
- Sorting these distances requires $O(h^2\log(h))$.

* Available at: https://www.scipy.org.

The total complexity for routing a single cluster is $O(h^2 \log(h) + h^2) = O(h^2 \log(h))$. For K clusters, the overall complexity of the routing phase is $O(K \cdot h^2 \log(h))$, which reduces to $O(n \log(n))$ when $h \approx \frac{n}{K}$.

Overall Complexity:
The clustering and routing phases are repeated until the centroids' displacement across iterations does not exceed a threshold ϵ. Since IT denotes the maximum number of iterations, the overall complexity of the two-phase CHB algorithm is then $O(n \times IT \log(n))$. □

5 Improving routes

To further improve the routes found by Algorithm 3, an ant colony optimization method in [3] is followed. The difference here is that the initial routes are the same routes found by CHB heuristic, rather than random ones. To be more precise, the amount of initial pheromones are not random numbers. The algorithm in [3] is provided to solve CVRP, while we apply it on any single route as TSP.

Ant colony optimization algorithm has several parameters that should be determined by user, and there is no deterministic certificate to show which value is the best. In different applications, different values may behave better. Table 2 introduces parameters of ant colony optimization algorithm.

Table 2: Parameters of ant colony optimization algorithm

Parameter	Definition	Parameter	Definition
$MaxIT$	Maximum no. of iterations	α	Pheromone importance
τ_0	Initial pheromone on each arc	β	Distance importance
η_{ij}	Inverse of d_{ij}	ρ	Evaporation coefficient
τ_{ij}	Amount of pheromone on arc (i, j)	r_0	A constant
d_{ij}	Distance between i and j	a	Index of ant, $a \in \{1, \dots, m\}$
E_P	Arcs of tour P		

It should be noted that the initial pheromone τ_0 is usually supposed to be the inverse of the best-known route distance found for that particular problem. Here τ_0 is evaluated in a way to strengthen the arcs in the tour found by Algorithm 3. So initially, we set $\tau_0 = 0.1$ for arcs not included in the tour and $\tau_0 = 1$ for arcs in it. Following steps explain ant colony method in details.

Algorithm 4 Ant colony optimization algorithm to improve routes of CHB heuristic

---

Input: Tour P starts and ends to depot.
Output: Improved tour $Best_{tour}$ starts and ends to depot.

Initialize parameters according to Table 2, $Best = \text{inf}$. ▷ (Initialization)
Let $\tau_{ij} = 1$, for all $(i, j) \in E_P$ and $\tau_{ij} = 0.1$ for all $(i, j) \notin E_P$.
for $Iter = 1, \ldots, MaxIT$ **do**
  for $a = 1, \ldots, m$ **do**
    Locate ant a in depot. Let $i = 0$. ▷ (The first location of ant a)
    $Tour = \{0\}$
    while $|Tath| < n + 1$ **do**
      Choose random number $r \in [0, 1]$ by uniform distribution.
      if $r \leq r_0$ **then**
        let $j = \underset{u \in Tour}{a} rg \max(\tau_{iu})^{\alpha}(\eta_{iu})^{\beta}$ ▷ (Choose the next node by instinct)
      else
        For any unvisited customer w, let $P_{iw} = \frac{(\tau_{iw})^{\alpha}(\eta_{iw})^{\beta}}{\sum_{u \notin Tour}(\tau_{iu})^{\alpha}(\eta_{iu})^{\beta}}$
        Choose next node j according to distribution function of P.
      end if
      Update $Tour := [Tour, j]$
      Update pheromone on arc (i, j) by $\tau_{ij} = (1 - \rho)\tau_{ij} + \rho\eta_{ij}$
      Let $i = j$ ▷ (Update current node)
    end while
    Let $Tour = [Tour, 0]$, L_a =sum of lengths of arcs in $Tour$.
    if $L_a < Best$ **then**
      $Best = L_a$ and $Best_{Tour} = Tour$
    end if
  end for
  for $(i, j) \in Best_{Tour}$ **do**
    $\tau_{ij} = (1 - \rho)\tau_{ij} + \rho \sum_{a=1}^{m} \Delta_{ij}^{a}$, where Δ_{ij}^{a} is obtained by (1) ▷ (Global pheromone update)
  end for
end for

The formula (1), globally updates the pheromone.

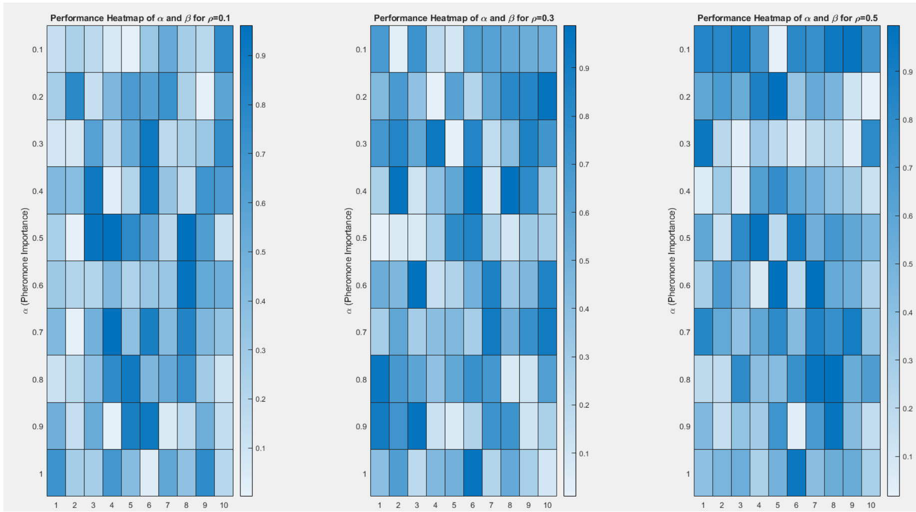

Figure 3: Performance heat map of α, β, and ρ.

$$\Delta_{ij}^{a} = \begin{cases} \frac{1}{L_a} & \text{If ant } a \text{ passes arc } (i,j) \\ 0 & \text{Otherwise} \end{cases} \tag{1}$$

The ant colony optimization algorithm may not change the solution by CHB heuristic, or even it may make it worse in some cases. However in most cases it improves the solution. There exists 3 parameters in this algorithm which should be initialized, α, performance importance, β, distance importance and ρ, evaporation coefficient. All test problems were solved with all values $\alpha \in \{0, 0.1, \ldots, 1\}$, $\beta \in \{1, 2, \ldots, 10\}$ and $\rho \in \{0.1, 0.3, 0.5\}$ for ten times and the average value of the final objective function is the performance of each triple (α, β, ρ). Figure 3 shows the results of these implementations. Note that the value of the performance is the value of the objective function, normalized between 0 and 1. Values 0 and 1 possess the lightest and the darkest color, respectively.

It can be seen that $(\alpha, \beta, \rho) = (0.1, 5, 0.1)$ provides the best average performance among all other choices. These values are fixed through all implementations.

6 Computational experiments

The proposed method is implemented in MATLAB 2020, 64 bit, and is run on Intel(R) Core(TM) i7-5500U CPU @ 2.4 GHz and 8 GB of RAM. We implement the proposed CHB heuristic in this way: First, nodes are clustered according to the CH-means algorithm 2, and find the respective routing by applying the CH-insertion algorithm 3 in any iteration. Afterwards, the best solution is improved by the ant colony algorithm 4 on CVRP benchmark problems from *CVRPlib*[**].
We apply the CH-means method with other insertion methods in literature. Furthermore, we apply the K-means method with the proposed CH-insertion method and compare the results. Table 3 explains all implemented approaches in this section.

Table 3: Implemented approaches for comparison

Clustering Method:	CH-means			K-means		
Routing Method: (insertion)	Clark & Wright insertion	Convex hull Nearest insertion	CH-insertion	Clark & Wright insertion	Convex hull nearest insertion	CH-insertion
Name:	CHmean-ClarkInsert	CHmean-NearestInsert	CHmean-CHInsert	Kmean-ClarkInsert	Kmean-NearestInsert	Kmean-CHInsert

The selection of K-means for comparison, is backed by its wide applicability in clustering problems. Two insertion methods, Clark & Wright method and convex hull nearest insertion method are also selected to be compared to CH-insertion method, for the sake of their speed and accuracy in compare with other insertion methods, [23]. All methods for TSP mentioned in [23] were tested on samples, and these methods resulted in best solutions among others. Therefore results show that Clark & Wright and convex hull nearest insertion methods are appropriate methods to be compared to the proposed CH-Insertion. First of all, we briefly explain the Clarke and Wright savings method and convex hull nearest insertion.

** available at: https://neo.lcc.uma.es/vrp/vrp-instances/capacitated-vrp-instances/

Clarke and Wright savings method

In the typical method in [12], savings $s_{ij} = c_{1i} + c_{1j} - c_{ij}$ for all customer points i and j are computed. Savings are ordered from largest to smallest. Initially, subtours $(0, i, 0)$ are formed for all customer points i. In any iteration, two subtours containing $(0, i)$ and $(j, 0)$, are merged, possessing the maximum amount of s_{ij} in the savings matrix. The new merged subtour contains (i, j). The procedure is repeated until all points are routed.

Convex hull nearest insertion

In this method, the convex hull of nodes in the given cluster is formed as an initial subtour, [55]. To each node d not yet contained, side (i, j) of the hull is assigned if minimizes $c_{di} + c_{dj} - c_{ij}$. Next, (i^*, d^*, j^*) is determined, which minimizes $(c_{i^*d^*} + c_{d^*j^*})/ci^*j^*$. Finally, node $d*$ is inserted in the subtour between nodes i^* and j^*. The procedure is repeated until all points of the cluster are routed.

We test the proposed CHB heuristic and the methods mentioned in Table 1, on three groups of benchmark problems: set A (Augerat, 1995), set E (Christofides and Eilon, 1969) and set P (Augerat, 1995). Tables 4, 5, and 6 show the results for set A, set B and set C, respectively. The last column of all tables show the optimal value. All optimal solutions and values are accessible from *CVRPlib*. The underlined instances are those with remained nodes without any enough capacity, left in vehicles as mentioned in Remark 1.

Based on the results in most instances, the proposed CHB heuristic is as good as other methods. The CHB heuristic, in some instances such as A-n44-k6 results in the worst cost, while in some other cases such as P-n76-k5 in both phases or in E-n101-k8 in routing phase, it achieves the best solution among other approaches.

Figure 4 shows the results more clearly. Any point in this figure shows the relative error of the corresponding method based on optimal value of the corresponding instance. The black line in left-hand figures is related to the

Table 4: Cost comparison of CVRP's instances: Set A

Benchmark instance	CHmean- Clark Insert	CHmean- Nearest Insert	CHmean- CH Insert	Kmean- Clark Insert	Kmean- Nearest Insert	Kmean- CH Insert	Optimal Value
A-n32-k5	934	928	928	872	872	872	784
A-n33-k6	834	834	834	807	807	807	742
A-n37-k6	1034	1034	1034	1029	1029	1029	949
A-n38-k5	825	825	825	853	853	856	730
A-n39-k6	857	857	857	905	905	905	831
A-n44-k6	1182	1114	1186	1011	1011	1011	937
A-n45-k7	1188	1188	1188	1245	1245	1245	1146
A-n46-k7	995	995	995	1025	1024	1024	914
A-n53-k7	1172	1171	1171	1109	1109	1190	1010
A-n60-k9	1623	1621	1625	1526	1529	1526	1354
A-n65-k9	1439	1437	1439	1415	1421	1415	1174
A-n69-k9	1350	1350	1351	1386	1388	1388	1159
A-n80-k10	2094	2089	2096	2000	1999	2039	1763

two-phase CHB method and the black line in right-hand figures is related to K-means with CH-insertion method. It is clear that black line manages to achieve the least relative error in some instances, both in right-hand and left-hand figures. This means that the proposed two-phase CHB heuristic is efficient in both predicting clusters and routing customers. Therefore, both proposed phases are efficient to be combined with other methods and also to be applied independently, as well.

Figure 5 also shows the percentage of obtaining the best cost solution among other approaches for different three sets A, E and P. In some cases two approaches result in the same solution, such as CH-means clustering with Clark routing and nearest insertion routing. This is shown by phrase "CHmean-Clark & NearestInsert" in the figure. Moreover, the phrase "CHmeans-Anycluster" means that all three methods for routing with CH-means clustering, managed to achieve the best solution among other approaches. Based on Figure 5, the proposed CH-means method is successful in instances of set A and P, while CH-insertion method behaves efficiently in

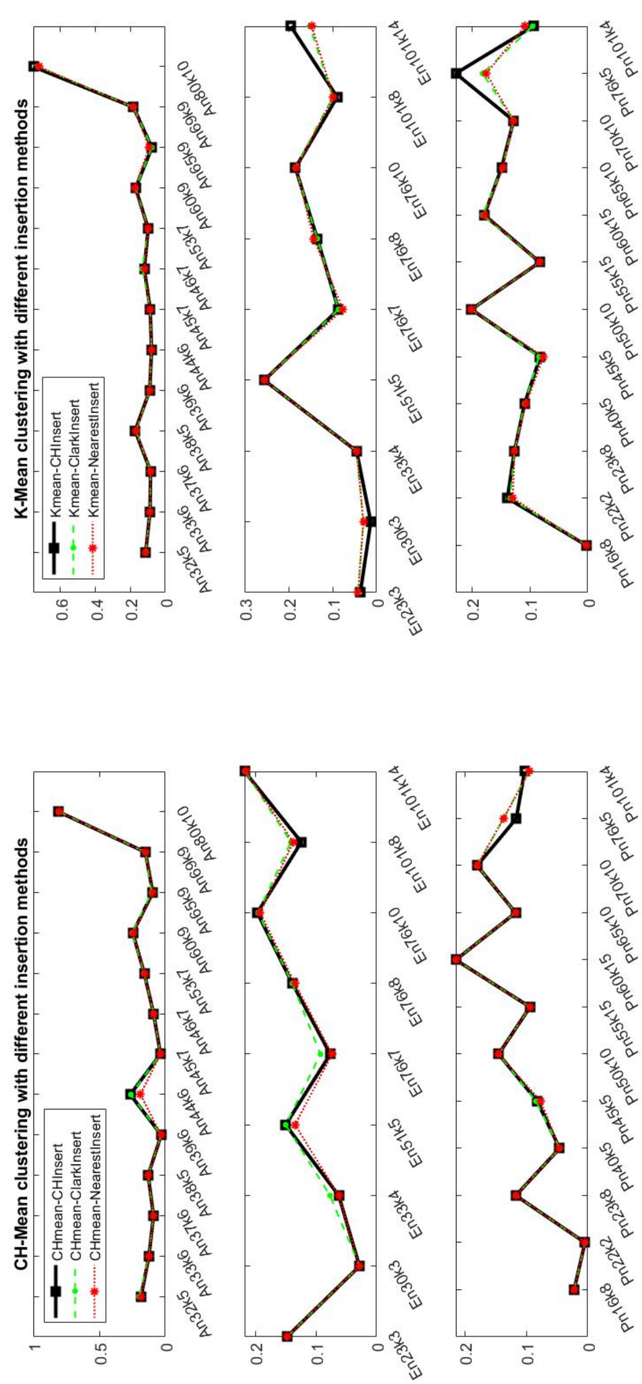

Figure 4: Comparison of relative error of cost different approaches according to optimal value, divided by clustering methods.

Table 5: Cost comparison of CVRP's instances: Set E

Benchmark instance	CHmean- Clark Insert	CHmean- Nearest Insert	CHmean- CH Insert	Kmean- Clark Insert	Kmean- Nearest Insert	Kmean- CH Insert	Best Found
E-n23-k3	653	653	653	592	592	592	569
E-n30-k3	549	549	549	550	550	547	534
E-n33-k4	889	887	887	873	873	873	835
E-n51-k5	599	591	599	654	654	654	521
E-n76-k7	735	732	734	741	734	741	682
E-n76-k8	837	834	837	834	835	834	735
E-n76-k10	992	989	993	983	982	983	830
E-n101-k8	930	927	926	896	895	891	815
E-n101-k14	1298	1297	1299	1229	1224	1275	1067

instances of set E.

Finally, Figure 6 compares the proposed CH-means method to the K-means method. Generally, it can be seen that the CH-means clustering methods is better in instances of set E, while in other two sets its vice versa.

7 Conclusion

In this paper, a two-phase CHB heuristic method has been introduced for CVRP, with computational complexity of $o(n^2 \log n)$, where n is the number of customers. Customers are first clustered by a CH-means algorithm. In a typical K-means algorithm, points are clustered in order to minimize the total dissimilarity between the points in a cluster. In CVRP, the length of the whole TSP tour should be minimized, because finally a TSP will be solved in any cluster. Hence, unlike typical K-means algorithm, in CH-means algorithm, initial centers are set in equidistant locations inside the region surrounded by customers. Moreover, the center of each cluster is updated according to the convex hull of the points assigned to that cluster. The

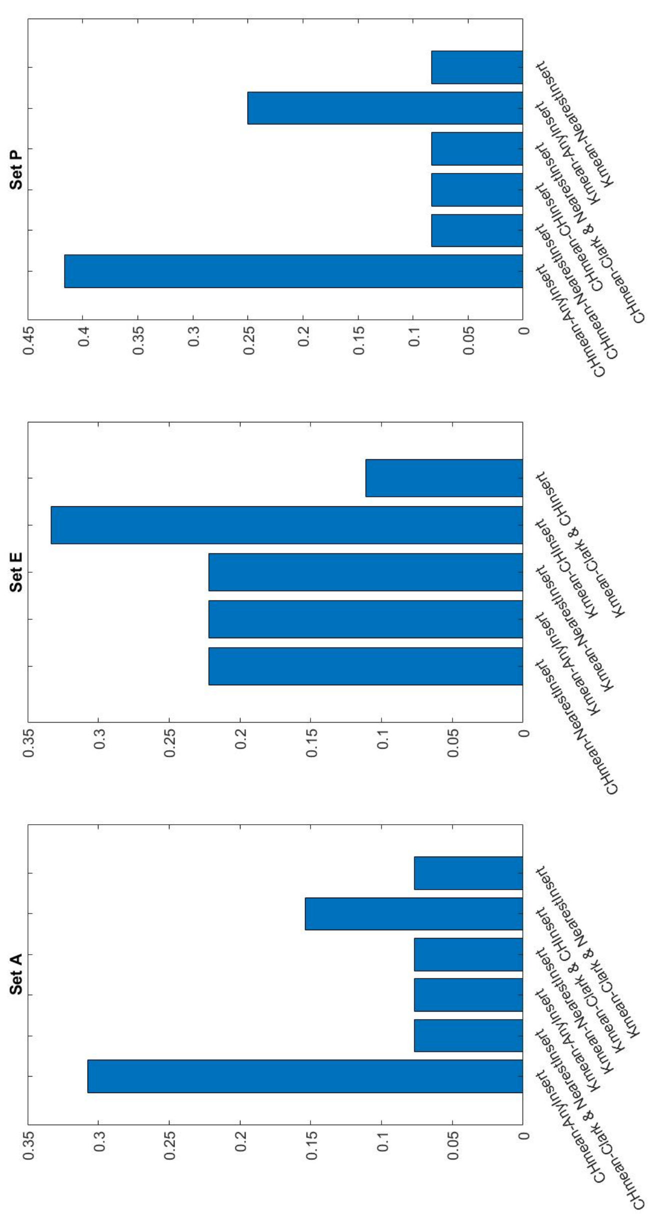

Figure 5: Comparison of relative error of cost different approaches according to optimal value, divided by clustering methods.

Table 6: Cost comparison of CVRP's instances: Set P

Benchmark instance	Clark Insert	CHmean- Nearest Insert	CH Insert	Clark Insert	Kmean- Nearest Insert	CH Insert	Best Found
P-n16-k8	460	460	460	451	451	451	450
P-n22-k2	217	217	217	245	244	246	216
P-n23-k8	591	591	591	596	596	596	529
P-n40-k5	479	479	479	508	507	508	458
P-n45-k5	553	549	552	553	549	552	510
P-n50-k10	797	797	797	836	836	836	696
P-n55-k15	1082	1082	1082	1071	1071	1071	989
P-n60-k15	1176	1176	1176	1144	1143	1143	968
P-n65-k10	885	885	885	909	908	910	792
P-n70-k10	976	976	976	933	933	933	827
P-n76-k5	713	713	712	741	737	770	627
P-n101-k4	746	746	751	746	755	749	681

new center is the mean point of that convex hull. The reason of this choice for updating centroids, is to make any cluster contain all points on the line segment joining any two points in it. However, adding points in any step is the same with the K-means algorithm and is based on the minimum distance from center, but since closest points to the center are located inside the respective convex hull, they will be assigned to that cluster.

Compared to the typical two-phase algorithms for CVRP, the novelty of CHB heuristic is that the fitness of any step's solution is calculated by the CHB routing algorithm, which is called CH-insertion, and it is saved. At the end of the clustering phase, the best found solution is chosen for improvement, which is not necessarily the last solution found. Although this increases the computational complexity of the algorithm, but it allows the selection of the best clustering.

The routes are built through the CH-insertion method. The idea for routing is to construct the convex hull of the points and to insert unassigned points by breaking an edge of the polygon into two edges. This is based on the well-known property for Euclidean TSP by [30]. Finally, the routes are im-

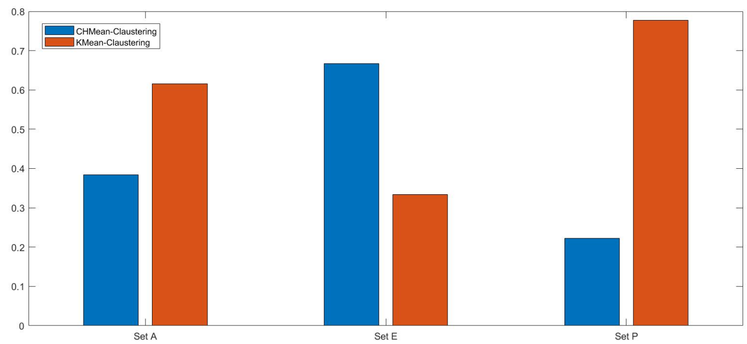

Figure 6: Percentage of achieving the best solution in approaches by CH-means clustering and K-means.

proved with a meta-heuristic algorithm. Ant colony optimization algorithm has been chosen, by virtue of its satisfactory performance on TSP.

Implementing the CHB heuristic on benchmark samples and comparing it to other two-phase heuristics show that the proposed two-phase CHB heuristic is efficient in both clustering -even if combined with other routing methods- and routing -even if combined with other clustering methods. Moreover, the proposed CHB heuristic results in the best solution among other implemented methods in this paper.

References

[1] Baldacci, R., Christofides, N. and Mingozzi, A. *An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts*, Math. Program., 115 (2008), 351–385.

[2] Barreto, S., Ferreira, C., Paixao, J. and Santos, B.S. *Using clustering analysis in a capacitated location-routing problem*, Eur. J. Oper. Res., 179 (2007), 968–977.

[3] Bell, J. and McMullen, P. *Ant colony optimization techniques for the vehicle routing problem*, Adv. Eng. Inform., 18 (2004), 41–48.

[4] Bodin, L.D. and Golden, B.L. *Classification in vehicle routing and scheduling*, Networks, 11 (1981), 97–108.

[5] Bodin, L.D. *The state of the art in the routing and scheduling of vehicles and crews*, Urban Mass Transp. Adm., Vol. 1, 1983.

[6] Bruwer, F. *Petal-shaped clustering for the capacitated vehicle routing problem*, Master Thesis, Univ. Witwatersrand, Johannesburg, 2018.

[7] Buhrmann, J.H., Campbell, I. and Ali, M. *A capacitated clustering heuristic for large datasets*, Proc. Int. Conf. Ind. Eng. Oper. Manag., 2018.

[8] Carlsson, J., Ge, D., Subramaniam, A., Wu, A. and Ye, Y. *Solving min-max multi-depot vehicle routing problem*, Lect. Glob. Optim., 55 (2009), 31–46.

[9] Chen, A., Gu, X. and Gao, Z. *Two-phase algorithm to multiple depots vehicle routing problem with soft time windows*, IOP Conf. Ser.: Earth Environ. Sci., 2020.

[10] Christofides, N., Mignozzi, A. and Toth, P. *The vehicle routing problem*, In: Christofides, N. (ed.), Combinatorial Optimization, Wiley, Hoboken (1979), 315–338.

[11] Cinar, D., Gakis, K. and Pardalos, P.M. *A 2-phase constructive algorithm for cumulative vehicle routing problems with limited duration*, Expert Syst. Appl., 56 (2016), 48–58.

[12] Clarke, G. and Wright, J.R. *Scheduling of vehicles from a central depot to a number of delivery points*, Oper. Res., 12(4) (1964), 568–581.

[13] Comert, S., Yazgan, H., Sertvuran, I. and Sengul, H. *A new approach for solution of vehicle routing problem with hard time window: An application in a supermarket chain*, Indian Acad. Sci., 42(12) (2017), 2067–2080.

[14] Dang, S., Liu, Y., Luo, Z., Liu, Z. and Shi, J., *Survey of the routing problem for cooperated trucks and drones.*, Drones, 8(10) (2024), 550.

[15] Dantzig, G. and Ramser, J., *The truck dispatching problem*, Manag. Sci., 6(1) (1959), 80–91.

[16] Dewinter, M., Vandeviver, Ch., Vander Beken, T. and Witlox, F., *Analyzing the police patrol routing problem: A review*, ISPRS Int. J. Geo-Inf., 9(3) (2020), 157.

[17] Ding, N., Li, M. and Hao, J., *A two-phase approach to routing a mixed fleet with intermediate depots*, Mathematics, 11(8) (2023), 1924.

[18] Dondo, R. and Cerda, J., *A cluster-based optimization approach for the multi-depot heterogeneous fleet vehicle routing problem with time windows*, Eur. J. Oper. Res., 176 (2007), 1478–1507.

[19] Euchi, J. and Saduk, A., *Hybrid Genetic-Sweep algorithm to solve the vehicle routing problem with drones*, Phys. Commun., 44 (2020), 101236.

[20] Fatemi-Anaraki, S., Mokhtarzadeh, M., Rabbani, M. and Abdolhamidi, D., *A hybrid of K-means and genetic algorithm to solve a bi-objective green delivery and pick-up problem*, J. Ind. Prod. Eng., (2021), 1–12.

[21] Fisher, M.L. and Jaikumar, R., *A generalized assignment heuristic for vehicle routing*, Networks, 11(2) (1981), 109–124.

[22] Gillett, B. and Miller, L., *A heuristic algorithm for the vehicle-dispatch problem*, Oper. Res., 22(2) (1974), 340–349.

[23] Golden, B., Bodin, L., Doyle, T. and Stewart, W., JR., *Approximate traveling salesman algorithms*, Oper. Res., 28(3), part 2 (1979), 694–711.

[24] Goutham, M., Menon, M., Garrow, S. and Stockar, S., *A convex hull cheapest insertion heuristic for the non-euclidean TSP*. arXiv preprint arXiv:2302.06582, 2023.

[25] Hammami, F., Rekik, M. and Coelho, L.C., *A hybrid adaptive large neighborhood search heuristic for the team orienteering problem*, Comput. Oper. Res., 123 (2020), 105034.

[26] Hertrich, C., Hungerländer, P. and Truden, C., *Sweep algorithms for the capacitated vehicle routing problem with structured time windows*, Oper. Res. Proc. 2018, (2019), 127–133.

[27] Imran, N. and Won, M., *VRPD-DT: Vehicle routing problem with drones under dynamically changing traffic conditions*, arXiv preprint arXiv:2404.09065 (2024).

[28] Jones, J. and Adamatzky, A., *Computation of the traveling salesman problem by a sShrinking blob*, Nat. Comput., 13(1) (2014), 1–16.

[29] Laporte, G., and Nobert, Y., *A branch and bound algorithm for the capacitated vehicle routing problem*, Oper.-Res.-Spektrum, 5(2) (1983), 77–85.

[30] Larson, R. and Odoni, A., *Urban operations research*, Prentice Hall, Englewood Cliffs, N.J., 1981.

[31] Liao, T.-Y., *On-line vehicle routing problems for carbon emissions reduction*, Comput.-Aided Civ. Infrastruct. Eng., 32(12) (2017), 1047–1063.

[32] Liu, Q., Xu, P., Wu, Y., and Shen, T., *A two-stage algorithm for vehicle routing problem with charging relief in post-disaster*, IET. Intel. Transp. Sys., 45(3) (2023), 123–137.

[33] Luo, J., and Chen, M.R., *Multi-phase modified shuffled frog leaping algorithm with extremal optimization for the MDVRP and the MDVRPTW*, Comput. Ind. Eng., 72 (2014), 84–97.

[34] Mahmud, N., and Haque, M.M., *Solving multiple depot vehicle routing problem (MDVRP) using genetic algorithm*, 2019 Int. Conf. Electr. Comput. Commun. Eng., (2019), 1–6.

[35] Marinakis Y., Marinaki M. and Migdalas A., *Variants and formulations of the vehicle routing problem*, Open Probl. Optim. Data Anal., Springer Optim. Its Appl., (2018), 91–127.

[36] Meira, L., Martins, P., Menzori, M. and Zeni, G., *Multi-objective vehicle routing problem applied to large scale post office deliveries*, arXiv preprint arXiv:1801.00712, (2017).

[37] Nadizadeh, A., Sahraeian, R., Sabzevarizadeh, A. and Homayouni, S.M., *Using greedy clustering method to solve capacitated location-routing problem*, Afr. J. Bus. Manag., 5(17) (2011), 7499–7506.

[38] Nallusamy, R., Duraiswamy, K., Dhalanaksmi, R. and Parthiban, P., *Optimization of multiple vehicle routing problems using approximation algorithms*, Int. J. Eng. Sci. Technol., 1(3) (2009), 129–135.

[39] Narasimha, K., *Ant colony optimization technique to solve min-max multi depot vehicle routing problem*, Master Thesis, Natl. Inst. Technol. Karnataka, India, 2011.

[40] Negreiros, M. and Palhano, A., *The capacitated centred clustering problem*, Comput. Oper. Res., 33(6) (2006), 1639–1663.

[41] Nuriyeva, F. and Kutucu, H., *A convex hull based algorithm for solving the traveling salesman problem*, TWMS J. App. Eng. Math. (2025).

[42] Nurkahyo, G., Alias, R., Shamsuddin, S.M. and Noor, Md., *Sweep algorithm in vehicle routing problem for public transport*, J. Antarabangsa, 2 (2002), 51–64.

[43] Poot, A., Kant, G. and Wagelmans, A., *A savings based method for real-life vehicle routing problems*, J. Oper. Res. Soc., 53 (2002), 57–68.

[44] Pourmohammadreza, N. and Jokar, M.R.A., *A novel two-phase approach for optimization of the last-mile delivery problem with service options*, Sustainability, 15(10), (2023), 8098.

[45] Ralphs, T.K., Kopman, L., Pulleyblank, W.R. and Trotter, L.E., JR., *On the capacitated vehicle routing problem*, Math. Program., 94 (2003), 343–359.

[46] Renaud, J., Laporte, G. and Boctor, F., *A Tabu search heuristic for the multi-depot vehicle routing problem*, Comput. Oper. Res., 23(3) (1996), 229–235.

[47] Rossit, D., Vigo, D., Tohme, F. and Frutos, M., *Improving visual attractiveness in capacitated vehicle routing problems: A heuristic algorithm*, XVIII Lat.-Iberoam. Conf. Oper. Res.-CLAIO, 749, 2016.

[48] Sariklis, D. and Powell, S., *A heuristic method for the open vehicle routing problem*, J. Oper. Res. Soc., 51(5) (2000), 564–573.

[49] Sehta, N. and Thakar, U., *Capacitated vehicle routing problem: A solution using convex hull based sweep algorithm and genetic algorithm*, In AIP Conference Proceedings (Vol. 2705, No. 1). AIP Publishing, 2023.

[50] Simchi-Levi, D. and Bramel, J., *The logic of logistics: Theory, algorithms and applications for logistics management*, Springer-Verlag, New York, 2nd ed., 1997.

[51] Singanamala, P., Reddy, K. and Venkataramaiah, P., *Solution to a multi depot vehicle routing problem using K-means algorithm, Clarke and Wright algorithm and ant colony optimization*, Int. J. Appl. Eng. Res., 13(21) (2018), 15236–15246.

[52] Solomon, M., *Algorithms for the vehicle routing and scheduling problems with time window constraints*, Oper. Res., 35(2) (1987), 254–265.

[53] Solomon, M.M. and Desrosiers, J., *Time window constrained routing and scheduling problems*, Transp. Sci., 22(1) (1988), 1–13.

[54] Soto-Concha, R., Escobar, J.W., Morillo-Torres, D. and Linfati, R., *The vehicle-routing problem with satellites utilization: A systematic review of the literature.*, Mathematics, 13(7) (2025), 1–29.

[55] Stewart, W.R., *Computational comparison of five heuristic algorithms for the Euclidean traveling salesman problem*, Lect. Notes Econ. Math. Syst., 199 (1982), Springer, Berlin, Heidelberg.

[56] Thibbotuwawa, A., Bocewicz, G., Nielsen, P. and Banaszak, Z., *Unmanned aerial vehicle routing problems: A literature review*, Appl. Sci., 10(13) (2020), 4504.

[57] Toth, P. and Vigo, D., *Branch-and-bound algorithms for the capacitated VRP*, SIAM, (2002), 29–51.

[58] Toth, P. and Vigo, D., Eds., *Vehicle routing: Problems, methods, and applications*, SIAM, 2014.

[59] Wang, Z. and Sheu, J.B., *Vehicle routing problem with drones*, Transp. Res. Part B Methodol., 122 (2019), 350–364.