2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

# A Hybrid Approach based on Genetic Algorithms for Solving the Clustered Vehicle Routing Problem

Petrică Pop, Camelia Chira

***Abstract*— In this paper, we describe a hybrid approach based on the use of genetic algorithms for solving the Clustered Vehicle Routing Problem, denoted by CluVRP. The problem studied in this work is a generalization of the classical Vehicle Routing Problem (VRP) and is closely related to the Generalized Vehicle Routing Problem (GVRP). Along with the genetic algorithm, we consider a local-global approach to the problem that is reducing considerably the size of the solutions space. The obtained computational results point out that our algorithm is an appropriate method to explore the search space of this complex problem and leads to good solutions in a reasonable amount of time.**

## I. Introduction

The classical Vehicle Routing Problem can be generalized in a natural way by considering a related problem relative to a given partition of the nodes of the graph into node sets (clusters), while the feasibility constraints are expressed in terms of the clusters. This generalized problem belongs to the class of generalized network design problems, known as well as generalized combinatorial optimization problems. For more information on generalized network design problems we refer to [1].

In the literature, there are considered three versions of the problem:

- one in which we are interested in designing optimally delivery or collection routes, from a given depot to a number of predefined, mutually exclusive and exhaustive clusters (node sets), visiting exactly one node from each cluster and subject to capacity restrictions. This problem is called the generalized vehicle routing problem (GVRP) and have been introduced by Ghiani and Improta [2].
- the second one is the problem of designing the optimally delivery or collection routes including at least one vertex from each cluster. This version of the problem was introduced by Baldacci et al. [3].
- the third one is the problem of designing the optimally delivery or collection routes such that all the nodes of each cluster must be visited consecutively. This problem is called the clustered vehicle routing problem (CluVRP) and have been introduced by Sevaux and Sörensen [4].

Petrică Pop is with the North University Center of Baia Mare, Technical University of Cluj-Napoca, Romania (email: petrica.pop@ubm.ro).

Camelia Chira is with Computer Science Department, Technical University of Cluj-Napoca, Romania (email: cchira@gmail.com).

This work was supported by a grant of the Romanian National Authority for Scientic Research, CNCS - UEFISCDI, project number PN-II-RU-TE-2011-3-0113.

From the described three versions, GVRP has generated a considerable interest in the last period, especially due to its practical applications. Efficient transformations of the GVRP into classical combinatorial optimization problems, for which exist heuristics, approximation algorithms or optimal solution methods, have been developed: Ghiani et al. [2] considered an transformation of the GVRP into Capacitated Arc Routing Problem (CARP), while Pop et al. [5] considered an efficient transformation of the GVRP into classical VRP. Integer programming formulations have been developed by Pop et al. [6]: a so called node formulation and a flow based formulation and Bektas et al. [7]: two based on multicommodity flow and the other two based on exponential sets of inequalities. The latter authors have proposed as well some branch-and-cut algorithms based on two of their models. The difficulty of obtaining optimum solutions for the GVRP has led to the development of some metaheuristic approaches. The first such algorithms were: a genetic algorithm based heuristic developed by Pop et al. [8], an adaptive large neighborhood search proposed by Bektas et al. [7], an incremental Tabu Search developed by Moccia et al. [9], a hybrid algorithm using a powerful local search procedure was described by Pop et al. [10].

The current literature on clustered vehicle routing problem is rather scarce: the initial motivation was a real world application involving parcel deliveries presented by Sevaux and Sörensen [4], a Simulated Annealing algorithm was described by Barthelemy et al [11] and Pop et al. [6] extended the integer programming formulations of the GVRP to CluVRP. Recently, Battarra et al. [12] described two exact solution algorithms and presented an application of the problem in the context of solid waste collection in urban areas. The same authors showed how is possible to transform an instance of CluVRP into an equivalent GVRP.

In this paper we confine ourselves to the clustered vehicle routing problem. The aim of the paper is to develop an efficient hybrid approach based on genetic algorithms for solving the CluVRP.

The remainder of this article is organized as follows: in Section II we provide the formal definition of the CluVRP, in Section III we describe in detail the components of our hybrid approach based on genetic algorithms and Section IV presents the obtained computational results. Finally, Section V concludes our work and provides some future work directions.

## II. Definition of the Clustered Vehicle Routing Problem

In this section we give a formal definition of the Clustered Vehicle Routing Problem as a graph theoretic model. Let $G = (V, A)$ be a directed graph with $V = \{0, 1, 2, ...., n\}$ as the set of vertices and the set of arcs

$$A = \{(i, j) \mid i, j \in V, i \neq j\}.$$

We have two kinds of arcs: the set intra-cluster arcs is defined by the vertices belonging to the same clusters and the set of inter-cluster arcs defined by vertics belonging to different clusters. The graph $G$ must be strongly connected and in general it is assumed to be complete.

Vertices $i \in \{1, ..., n\}$ correspond to the customers and the vertex $0$ corresponds to the depot. The entire set of vertices is partitioned into $k + 1$ mutually exclusive nonempty subsets, called clusters, $V_0, V_1, ..., V_k$, i.e. the following conditions hold:

1. $V = V_0 \cup V_1 \cup ... \cup V_k$
2. $V_l \cap V_p = \emptyset$ for all $l, p \in \{0, 1, ..., k\}$ and $l \neq p$.

The cluster $V_0$ has only one vertex $0$, which represents the depot, and remaining $n$ vertices are belonging to the remaining $k$ clusters.

A nonnegative cost $c_{ij}$ is associated with each arc $(i, j) \in A$ and represents the travel cost spent to go from vertex $i$ to vertex $j$.

Each customer $i$ ($i \in \{1, ..., n\}$) is associated with a known nonnegative demand $d_i$ to be delivered and the depot has a fictitious demand $d_0 = 0$. Given a cluster $V_p \subset V$, let $d(V_p) = \sum_{i \in V_p} d_i$ the total demand of the cluster $V_p$, $p \in \{1, ..., k\}$.

There exist $m$ identical vehicles, each with a capacity $Q$ and to ensure feasibility we assume that $d_i \leq Q$ for each $i \in \{1, ..., n\}$. Each of the vehicles may perform at most one route.

The *clustered vehicle routing problem* (CluVRP) consists in finding a collection of simple circuits (each corresponding to a vehicle route) visiting all the clusters with minimum cost, defined as the sum of the costs of the arcs belonging to the circuits and such that the following constraints hold:

i) each circuit starts and ends at the depot vertex;
ii) all the vertices of each cluster must be visited consecutively by a circuit;
iii) the sum of the demands of the visited vertices by a circuit does not exceed the capacity of the vehicle, $Q$.

An illustrative scheme of the CluVRP and a feasible tour is shown in the Figure 1.

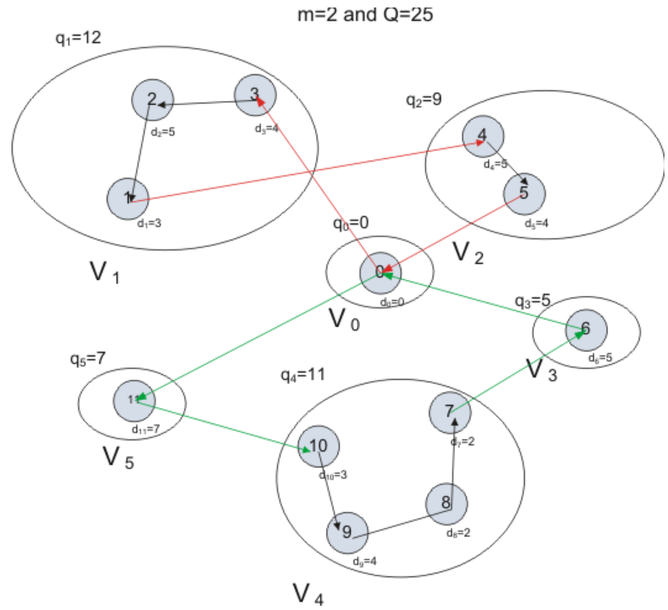


Fig. 1. A feasible solution of the CluVRP

A feasible solution of the CluVRP consists of a collection of routes, each visiting the depot and all the vertices from each cluster consecutively. We will call such a route a *generalized clustered route*. The order of visiting the clusters will be called *global route*. In the example presented in Figure 1, $0-3-2-1-4-5-0$ and $0-11-10-9-8-7-6-0$ are generalized clustered routes and $V_0 - V_1 - V_2 - V_0$ and $V_0 - V_5 - V_4 - V_3 - V_0$ are global routes.

The CluVRP reduces to the classical capacitated Vehicle Routing Problem (VRP) when all the clusters are singletons.

The CluVRP is $NP$-hard because it includes the classical capacitated Vehicle Routing Problem as a special case when all the clusters are singletons.

## III. The hybrid approach based on Genetic Algorithms for solving the Clustered Vehicle Routing Problem

We present in this section a hybrid algorithm for solving the CluVRP obtained by combining a genetic algorithm with a local-global approach to the problem.

### A. The local-global approach to the Clustered Vehicle Routing Problem

Based on the definition of the generalized combinatorial optimization problems, a natural approach that takes advantages between them and their classical variants, is the local-global approach introduced by Pop [1] for the first time in the case of the generalized minimum spanning tree problem. This original approach opened new directions of research, several exact, heuristic, metaheuristic and hybrid algorithms being proposed based on it for several generalized combinatorial optimization problems.

The local-global approach aims at distinguishing between *global connections* (connections between clusters) and *local connections* (connections between nodes belonging to different clusters).

We denote by $G^g = (V^g, A^g)$ the graph obtained from $G$ after replacing all the nodes of a cluster $V_i$ with a supernode representing $V_i$, $\forall i \in \{1, ..., k\}$, the cluster $V_0$ (depot) consists already of one node. We will call the graph

$G^g$ the *global graph*. In this graph the set of nodes is $V^g = \{V_0, V_1, ..., V_k\}$ and the set of arcs

$$A^g = \{(V_i, V_j) \mid \exists(u, v) \in A \wedge u \in V_i \wedge v \in V_j\}.$$

Figure 2 presents the global routes associated to the feasible solution shown in Figure 1.

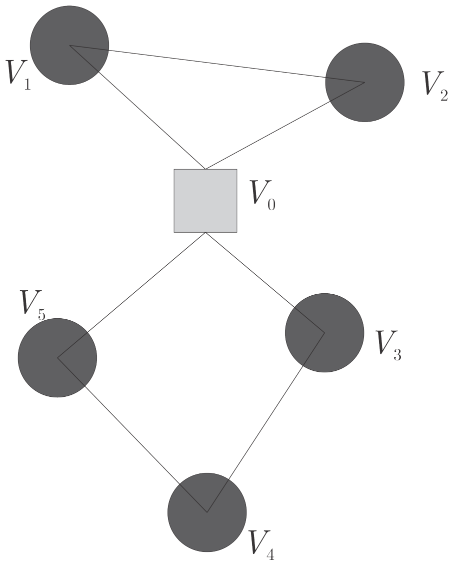


Fig. 2. An example of global routes

We consider now a feasible solution in the global graph, i.e. a collection of $r$ global routes of form $(V_0, V_{k_1}, ..., V_{k_p})$ in which the clusters are visited. Each global route on $G^g$ represents the set of all feasible generalized clustered routes on $G$ which contains for each arc $(V_{k_l}, V_{k_{l+1}}) \in A^g$ a path of the form $(i_1, i_2, ..., i_t, j)$ with $i_1, i_2, ..., i_t \in V_{k_l}$ and $j \in V_{k_{l+1}}$. Such a set of generalized clustered routes on G that a particular global route represents is in general exponentially large with respect to the number of nodes.

Our aim to find the best feasible solution of the CluVRP (w.r.t. cost minimization), i.e. a collection of $r$ generalized clustered routes visiting the clusters according to the given sequence.

In order to show a generalized clustered route visiting the clusters according to a given sequence $(V_0, V_{k_1}, ..., V_{k_p})$ we construct a layered network (LN) with $p + 2$ layers corresponding to the clusters $V_0, V_{k_1}, ..., V_{k_p}$ and in addition we duplicate the cluster $V_0$.

The layered network contains all the nodes of the clusters $V_0, V_{k_1}, ..., V_{k_p}$ plus an extra node $0' \in V_0$ and the arcs are defined as follows: there is an arc $(0, i)$ for each $i \in V_{k_1}$ with the cost $c_{0i}$, an arc $(i_u, i_v)$ for each $i_u, i_v \in V_{k_l}$, $l \in \{1, ..., p\}$ with the cost $c_{i_u i_v}$, an arc $(i, j)$ for each $i \in V_{k_l}$ and $j \in V_{k_{l+1}}$ $(l = 1, ..., p-1)$ having the cost $c_{ij}$ and an arc $(i, 0')$ for each $i \in V_{k_p}$ having the cost $c_{i0'}$.

In Figure 3, we present the constructed layered network and we point out a generalized clustered route visiting the clusters according to the given sequence $(V_0, V_{k_1}, ..., V_{k_p})$.

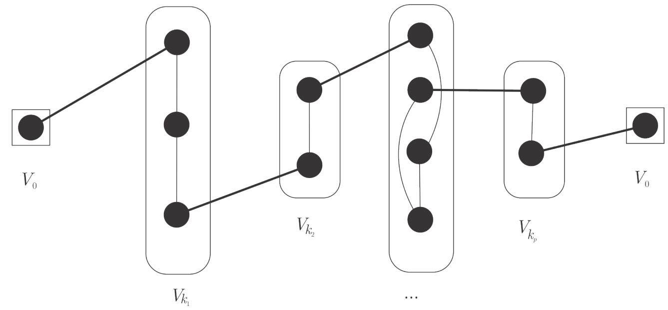


Fig. 3. Example showing a generalized clustered route visiting the clusters $V_0, V_{k_1}, ..., V_{k_p}$ in the constructed layered network LN

### *B. The Genetic Algorithm*

*1) Genetic Encoding:* In our algorithm we used the following genetic representation of the solution domain: an individual is represented as a list of clusters

$$(V_{l_1}^{(1)}, V_{l_2}^{(1)}, ..., V_{l_p}^{(1)}, V_0, ..., V_0, V_{l_1}^{(r)}, V_{l_2}^{(r)}, ..., V_{l_t}^{(r)})$$

representing a collection of $r$ global routes $V_0 - V_{l_1}^{(1)} - V_{l_2}^{(1)} - ... - V_{l_p}^{(1)} - V_0$,..., $V_0 - V_{l_1}^{(r)} - V_{l_2}^{(r)} - ... - V_{l_t}^{(r)} - V_0$, where $p, t \in N$ with $1 \leq p, t \leq k$.

For example in the case of Figure 1, an individual is: $(1\ 2\ 0\ 5\ 4\ 3)$ and represents the collection of 2 global routes, which is passing through the clusters in the following order:

$$(V_0\ V_1\ V_2\ V_0\ V_5\ V_4\ V_3\ V_0).$$

The values $\{1, ..., 5\}$ represent the clusters while the depot denoted by $\{0\}$ is the route splitter. The number of route splitters needed for $m$ vehicles is $m - 1$. Besides the depot stops added by the splitters in the solution, the first and last route receive the depot in the first and last position respectively in the global routes. In the example presented, route 1 begins at the depot then visits the clusters $V_1 - V_2$ and returns to the depot. Route 2 starts at the depot and visits the clusters $V_5 - V_4 - V_3$ returning then to the depot.

The collection of global routes needs several collections of generalized routes which are created as follows: for each cluster, the shortest path between any two nodes is determined considering solely the distance between the nodes in the cost function; next, the best (in terms of minimizing the total distance) from these shortest paths is selected specifying in this way a starting node and a terminal node to enter and respectively exit each cluster. These nodes complete the layered network described in the previous section and are further needed in determining the overall cost for all generalized clustered routes. The following three distance types will contribute to the fitness:

(i) the distance between the depot and the starting node of a cluster,
(ii) the distance between the terminal node of a cluster and the starting node of the next cluster, and

(iii) the distance between the terminal node of a cluster and the depot.

The example represented in Figure 1 uses the following pairs of starting, terminal nodes for each cluster: $3, 1 \in V_1$, $4, 5 \in V_2$, $6, 6 \in V_3$, $10, 7 \in V_4$ and $11, 11 \in V_5$. These starting and terminal nodes correspond to the best with regard to cost minimization for the collection of generalized routes.

We can see that our described representation allows empty routes by simply placing two route splitters together without clients between them. Some routes in the chromosome may cause the vehicle to exceed its capacity. When this happens, in order to guarantee that the interpretation is always a valid candidate solution, we perform the following modification: the route that exceeds the vehicle capacity is split at the cluster that causes the violation of capacity restrictions and the rest of clusters in that route are distributed among the other routes at random as long as all restrictions are met.

*2) The fitness value:* The fitness function is defined over the genetic representation and measures the quality of the represented solution. In our case, the fitness value of a feasible solution, i.e. a collection of global routes, is given by the cost of the best corresponding collection of generalized routes (with regard to cost minimization) obtained by constructing the layered network and determined by solving a given number of shortest path problems. For each cluster, the shortest path between any two nodes is predetermined in an exact way. From these shortest paths of each cluster, the best one in terms of distance is selected as the local route inside the cluster.

*3) Initial population:* The construction of the initial population is of great importance to the performance of GA, since it contains most of the material the final best solution is made of. Experiments were carried out with the initial population generated randomly and with an initial population of structured solutions. In order to generate the population of structured solutions we used a Monte Carlo based method. However, from the experiments carried out it turned out that the Monte Carlo method of generating the initial population has not brought any improvements. The initial population generated randomly having the advantage that is representative from any area of the search space.

*4) Genetic operators:* **Crossover**

The crossover operator combines two or more parents in order to generate the possibility of getting a better offspring.

Two parents are selected from the population by the binary tournament method. Offspring are produced from two parent solutions using the following 2-point order crossover procedure: it creates offspring which preserve the order and position of symbols in a subsequence of one parent while preserving the relative order of the remaining symbols from the other parent. It is implemented by selecting two random cut points which define the boundaries for a series of copying operations.

The recombination of two collections of global routes requires some further explanations. First, the symbols between the cut points are copied from the first parent into the first offspring. Then, starting from the first position, the symbols are copied from the second parent into the first offspring, omitting any symbols that were copied from the first parent and skipping the positions between the two cut points which are already filled. The second offspring is produced by swapping round the parents and then using the same procedure.

Next, we present the application of the proposed 2-point order crossover in the case of a problem consisting of 8 clusters and the depot. We assume two well-structured parents chosen randomly, with the cutting points between nodes 2 and 3, respectively 5 and 6:

$$\begin{array}{rcccccccccccc} P_1 = & 6 & 8 & | & 1 & 0 & 2 & | & 7 & 0 & 5 & 4 & 3 \\ P_2 = & 8 & 2 & | & 1 & 6 & 0 & | & 0 & 4 & 3 & 5 & 7 \end{array}$$

Note that the length of the two individuals is the same while the number of routes for individual $P_1$ is 3 (i.e. $0-6-8-1-0$, $0-2-7-0$ and $0-5-4-3-0$) and the number of routes for $P_2$ is only 2 (i.e. $0-8-2-1-6-0$ and $0-4-3-5-7-0$) due to the two consecutive positions occupied by the route splitter.

The sequences between the two cutting-points are copied into the two offspring:

$$\begin{array}{rcccccccccccc} O_1 = & x & x & | & 1 & 0 & 2 & | & x & x & x & x & x \\ O_2 = & x & x & | & 1 & 6 & 0 & | & x & x & x & x & x \end{array}$$

The nodes of the parent $P_1$ are copied into the offspring $O_2$ if $O_2$ does not contain already the clusters of $P_1$. If the current position in $P_1$ contains a route splitter then this is copied only if the offspring $O_2$ has not reached the maximum allowed number of splitters (i.e. an individual can not contain more routes than the available number of vehicles). Therefore, the offspring $O_2$ is:

$$\begin{array}{rcccccccccccc} O_2 = & 8 & 0 & | & 1 & 6 & 0 & | & 2 & 7 & 5 & 4 & 3 \end{array}$$

Then the nodes of the parent $P_2$ are copied into the offspring $O_1$ in the same manner. The nodes of the clusters not present in $O_1$ are copied into the remaining positions:

$$\begin{array}{rcccccccccccc} O_1 = & 8 & 6 & | & 1 & 0 & 2 & | & 0 & 4 & 3 & 5 & 7 \end{array}$$

The new generated individuals are validated against vehicle capacity restrictions and a similar repair procedure as the one performed in the population initialization is applied if necessary.

**Mutation**

We use in our genetic algorithm the following random mutation operator called the inter-route mutation operator which is a swap operator: it picks two random locations in the solution vector and swaps their values. Let the parent solution be $(6\ 8\ 1 \mid 0\ 2\ 7 \mid 0\ 5\ 4\ 3)$, then the inter-route mutation operator picks two random clusters, for example $V_8$ and $V_5$ and swaps their values obtaining the new chromosome:

$$(6\ 5\ 1 \mid 0\ 2\ 7 \mid 0\ 8\ 4\ 3).$$

*5) Selection:* Selection is the stage of a genetic algorithm in which individuals are chosen from a population to undergo a new generation. The selection process is deterministic.

In our algorithm we investigated and used the properties of $(\mu, \lambda)$ selection, where $\mu$ parents produce $\lambda$ $(\lambda > \mu)$ offspring and only the offspring undergo selection. In other words, the lifetime of every individual is limited to only one generation. The limited life span allows to forget the inappropriate internal parameter settings. This may lead to short periods of recession but it avoids long stagnation phases due to unadapted strategy parameters.

*6) Genetic parameters:* The genetic parameters are very important for the success of the algorithm, equally important as the other aspects, such as the representation of the individuals, the initial population and the genetic operators. The most important parameters are: the population size $\mu$ has been set to 10 times the number of clusters and the intermediate population size $\lambda$ was chosen twenty times the size of the population: $\lambda = 20 \cdot \mu$. Therefore, crossover was applied $\lambda$ times for two inidividuals randomly selected from the population of size $\mu$. Each offspring generated after crossover undergoes mutation with a probability of $20\%$. The number of generations used in our algorithm was set to 2000.

## IV. Numerical Experiments

Computational experiments are performed for several CluVRP instances which consider different number of nodes, clusters and vehicles which have been adapted from the 20 large-size CVRP instances described by Golden et al [14] and available at http://www.rhsmith.umd.edu/faculty/bgolden/vrp_data.htm.

Originally the set of nodes in these problems were not divided into clusters. Fischetti *et al.* [15] proposed in the case of the generalized traveling salesman problem a procedure to partition the nodes of the graph into clusters, called CLUSTERING. This procedure sets the number of clusters $s = [\frac{n}{\theta}]$, identifies the $s$ farthest nodes from each other and assigns each remaining node to its nearest center.

This procedure was used in order to adapt the CVRP instances to the CluVRP case with $\theta \in \{5, ..., 15\}$. The number of clusters in the corresponding CluVRP instances are ranging from 17 to 97, the total number of nodes are ranging from 241 to 481 and the number of nodes within a cluster are varying from 1 to 71.

However, the solution approach proposed in this paper is able to handle any cluster structure.

The testing machine was an Intel Dual-Core 1,6 GHz and 1 GB RAM. The operating system was Windows XP Professional. The algorithm was developed in Java, JDK 1.6.

The proposed evolutionary approach to CluVRP has been implemented and 30 runs of the algorithm based on the parameter setting given in the previous section have been performed. The best and the average solution from these runs are presented in Table 1. The first column in the table gives the name of the instances, the second column provides the number of the clusters and the third column contains the total number of nodes. The next two columns contains the values of the best solutions respectively the average solutions obtained using our hybrid genetic algorithm.

Analyzing the computational results, we observed that the best solutions were reached quite in the early phases of the generations, therefore we plan to maintain the diversity in order to avoid premature convergence by varying the population size and by changing the selection pressure. In addition we plan to use local search optimization in order to refine the solutions explored by the proposed hybrid genetic based algorithm.

TABLE I

THE EXPERIMENTAL RESULTS

| Pb. instance name | No. of clusters | No. of nodes | Best solution | Average solution |
|---|---|---|---|---|
| Golden1_C17_N241 | 17 | 241 | 5403.37 | 5425.32 |
| Golden1_C18_N241 | 18 | 241 | 5373.88 | 5409.30 |
| Golden1_C19_N241 | 19 | 241 | 5426.37 | 5461.08 |
| Golden1_C21_N241 | 21 | 241 | 5355.10 | 5396.02 |
| Golden1_C22_N241 | 22 | 241 | 5470.34 | 5506.15 |
| Golden1_C25_N241 | 25 | 241 | 5525.57 | 5585.07 |
| Golden1_C27_N241 | 27 | 241 | 5588.96 | 5631.01 |
| Golden1_C31_N241 | 31 | 241 | 5903.34 | 5971.26 |
| Golden1_C35_N241 | 35 | 241 | 6113.88 | 6199.17 |
| Golden1_C41_N241 | 41 | 241 | 6042.80 | 6163.40 |
| Golden1_C49_N241 | 49 | 241 | 5946.17 | 6066.17 |
| Golden2_C22_N321 | 22 | 321 | 8389.73 | 8419.28 |
| Golden2_C23_N321 | 23 | 321 | 8394.82 | 8434.97 |
| Golden2_C25_N321 | 25 | 321 | 8627.21 | 8695.93 |
| Golden2_C27_N321 | 27 | 321 | 8551.32 | 8664.17 |
| Golden2_C30_N321 | 30 | 321 | 8488.67 | 8605.71 |
| Golden2_C33_N321 | 33 | 321 | 8517.26 | 8668.53 |
| Golden2_C36_N321 | 36 | 321 | 8545.55 | 8689.74 |
| Golden2_C41_N321 | 41 | 321 | 8795.87 | 8918.46 |
| Golden2_C46_N321 | 46 | 321 | 9089.30 | 9266.23 |
| Golden2_C54_N321 | 54 | 321 | 9492.89 | 9646.24 |
| Golden2_C65_N321 | 65 | 321 | 9345.89 | 9574.47 |
| Golden3_C27_N401 | 27 | 401 | 11709.19 | 11780.66 |
| Golden3_C29_N401 | 29 | 401 | 11658.50 | 11857.91 |
| Golden3_C31_N401 | 31 | 401 | 11793.87 | 11928.24 |
| Golden3_C34_N401 | 34 | 401 | 11783.44 | 11959.50 |
| Golden3_C37_N401 | 37 | 401 | 11755.18 | 11977.78 |
| Golden3_C41_N401 | 41 | 401 | 11651.00 | 11912.38 |
| Golden3_C45_N401 | 45 | 401 | 11738.27 | 11996.54 |
| Golden3_C51_N401 | 51 | 401 | 11866.68 | 12209.38 |
| Golden3_C58_N401 | 58 | 401 | 12404.23 | 12708.65 |
| Golden3_C67_N401 | 67 | 401 | 12650.62 | 12980.77 |
| Golden3_C81_N401 | 81 | 401 | 12471.52 | 12898.42 |
| Golden4_C33_N481 | 33 | 481 | 15372.88 | 15608.98 |
| Golden4_C35_N481 | 35 | 481 | 15388.20 | 15575.91 |
| Golden4_C37_N481 | 37 | 481 | 15398.51 | 15571.08 |
| Golden4_C41_N481 | 41 | 481 | 15354.81 | 15621.26 |
| Golden4_C44_N481 | 44 | 481 | 15652.27 | 15865.84 |
| Golden4_C49_N481 | 49 | 481 | 15825.10 | 16110.58 |
| Golden4_C54_N481 | 54 | 481 | 15909.72 | 16275.88 |
| Golden4_C61_N481 | 61 | 481 | 15909.49 | 16351.40 |
| Golden4_C69_N481 | 69 | 481 | 16184.54 | 16632.87 |
| Golden4_C81_N481 | 81 | 481 | 16547.82 | 17026.52 |
| Golden4_C97_N481 | 97 | 481 | 16498.83 | 16977.09 |

## V. Conclusions

The Clustered Vehicle Routing Problem is an extension of the classical Vehicle Routing Problem and consists in finding a minimum cost collection of routes such that all the nodes from a given number of predefined, mutually exclusive and exhaustive clusters are visited consecutively.

We described an approach to the CluVRP based on distinguishing between global connections (connections between clusters) and local connections (connections between nodes from different clusters). Based on this approach to the problem, we presented a novel efficient hybrid algorithm. The proposed computational model to approach the problem is genetic algorithm applied with respect to the global graph, reducing in this way substantially the size of the solutions space.

In the future, we plan to combine our algorithm with a local search optimization procedure in order to refine the solutions explored by our hybrid GA. The potential benefits of an elitist strategy will also be investigated. In addition, we will need to asses the generality and scalability of the proposed hybrid heuristic by testing it on more instances and comparing it with other schemes such as Simulated Annealing and swarm intelligence techniques.

## REFERENCES


[1] P.C. Pop, *Generalized Network Design Problems. Modeling and Optimization.* De Gruyter Series in Discrete Mathematics and Applications, Germany, 2012.

[2] G. Ghiani, G. Improta, "An efficient transformation of the generalized vehicle routing problem," *European Journal of Operational Research*, vol. 122(1), pp. 11-17, 2000.

[3] R. Baldacci, G. Laporte, "Some applications of the generalized vehicle routing problem," *Journal of the Operational Research Society*, vol. 61(7), pp. 1072-1077, 2010.

[4] M. Sevaux, K. Sörensen, "Hamitonian paths in large clustered routing problems", in Proc. of EU/MEeting workshop on Metaheuristics for Logistics and Vehicle Routing, Troyes, France, 2008.

[5] P.C. Pop, C. Pop Sitar, "A new efficient transformation of the generalized vehicle routing problem into the classical vehicle routing problem", *Yugoslav Journal of Operations Research*, vol. 21(2), pp. 187-198, 2011.

[6] P.C. Pop, I. Kara, A. Horvat Marc, "New Mathematical Models of the Generalized Vehicle", *Applied Mathematical Modelling*, vol. 36(1), pp. 97-107, 2012.

[7] T. Bektas, G. Erdogan, S. Ropke "Formulations and Branch-and-Cut Algorithms for the Generalized Vehicle Routing Problem", *Transportation Science*, vol. 45(3), pp. 299-316, 2011.

[8] P.C. Pop, O. Matei, C. Pop Sitar, C. Chira, "A genetic algorithm for solving the generalized vehicle routing problem", *Lecture Notes in Artificial Intelligence*, vol. 6077, pp. 119-126, 2010.

[9] L. Moccia, J.-F. Cordeau, G. Laporte, "An incremental tabu search heuristic for the generalized vehicle routing problem with time windows", *Journal of the Operational Research Society*, vol. 63, pp. 232-244, 2012.

[10] P.C. Pop, O. Matei, C. Pop Sitar, "An improved hybrid algorithm for solving the generalized vehicle routing problem", *Neurocomputing*, vol. 109, pp. 76-83, 2013.

[11] T. Barthelemy, A. Rossi, M. Sevaux, K. Sörensen, "Metaheuristic approach for the clustered VRP", in Proc. of EU/ME, Lorient, France, 2010.

[12] M. Battarra, G. Erdogan, D. Vigo, "Exact algorithms for the clustered vehicle routing problem", submitted, 2013.

[13] J. Holland, *Adaptation in Natural and Artificial Systems.* University of Michigan Press, 1975.

[14] B.L. Golden, E.A. Wasil, J.P. Kelly and I.-M. Chao, *Metaheuristic in vehicle routing.* In Fleet Management and Logistics, pp. 33-56, Kluwer, Boston, 1998.

[15] M. Fischetti, J.J. Salazar and P. Toth, "A branch-and-cut algorithm for the symmetric generalized traveling salesman problem", *Operations Research*, vol. 45(3), pp. 378-394, 1997.