Soft Comput
DOI 10.1007/s00500-015-1970-4

METHODOLOGIES AND APPLICATION

Efficient route planning for an unmanned air vehicle deployed on a moving carrier

Halil Savuran[1] · **Murat Karakaya**[2]

Abstract Vehicle routing problem (VRP) is a constrained extension of the well-known traveling salesman problem (TSP). Emerging from the current conceptual trends in operations field, a new constraint to be included to the existing VRP parameters is the depot mobility. A practical example of such a problem is planning a route for an Unmanned air vehicle (UAV) deployed on a mobile platform to visit fixed targets. Furthermore, the range constraint of the UAV becomes another constraint within this sample case as well. In this paper, we define new VRP variants by introducing depot mobility (Mobile Depot VRP: MoDVRP) and extending it with capacity constraint (Capacitated MoDVRP: C-MoDVRP). As a sample use case, we study route planning for a UAV deployed on a moving carrier. To deal with the C-MoDVRP, we propose a Genetic Algorithm that is adapted to satisfy the constraints of depot mobility and range, while maximizing the number of targets visited by the UAV. To examine the success of our approach, we compare the individual performances of our proposed genetic operators with conventional ones and the performance of our overall solution with the Nearest Neighbor and Hill Climbing heuristics, on some well-known TSP benchmark problems, and receive successful results.

Communicated by V. Loia.

✉ Murat Karakaya
kmkarakaya@gmail.com; murat.karakaya@atilim.edu.tr

Halil Savuran
savuran.halil@student.atilim.edu.tr

1 Department of Software Engineering, Atilim University, Incek, Ankara, Turkey

2 Department of Computer Engineering, Atilim University, Incek, Ankara, Turkey

Keywords Carrier-deployed unmanned air vehicle · Range constraint · Genetic algorithm · Mobile depot · Vehicle routing problem

1 Introduction

One of the most prominent fields in which technology changes humans role is aviation. Especially, advanced technology removes constraints stemming from human in aviation by enabling unmanned air vehicles (UAVs). Even though their first appearance dates back as early as the 1920s, it is only since the late 1990s that UAVs have reached an advantageous and reliable level of technological maturity over manned aerial vehicles (Arjomandi et al. 2006). This development has led UAVs to be used for a wide variety of military and civilian purposes (Watts et al. 2012; Nonami 2007), and given rise to researches for new operational concepts aiming to exploit their capabilities. Utilizing carriers to augment their mobility is one of such capabilities (Sullivan et al. 2013; Martin and Dewolfe 2012; Pearson et al. 2006), and seeking effective routes for a UAV that takes off from and lands on a moving carrier constitutes the subject of this paper.

Regardless of whatever operational purpose it is tasked with, the mission life cycle of a UAV comes down to taking-off from a base, visiting defined targets to collect information, and returning back to a base. When we map elements such as UAV, targets and bases, respectively, to vehicle, customers and depots, the problem of effective route design for a carrier-launched UAV is categorically a VRP problem. Therefore, to study this problem, we model it as a variant of VRP with introducing the problem-specific constraints.

The constraints of depot mobility, range capacity, and the need for considering both at the same time bring out novel challenges for a VRP solution designer. Since the depot is mobile and its location changes in time, the route calculations will be different from that of a static depot VRP. Moreover, UAV has a flight range limitation and, thus, it must return to the moving platform before the UAV reaches its range. As the objective of the route planning is to visit as many as targets, these two constraints need to be handled at the same time in a solution to create a feasible and effective tour. Therefore, a possible solution would consider the following conditions to satisfy these constraints:

– The UAV should takeoff from a point on the given path of the carrier, such that it can visit more targets;
– The UAV should fly in such a route between the targets that more of them can be visited with less travel distance; and
– While doing these, the landing distance to the future location of the carrier should be included in the range calculations.

The ultimate purpose of this study is to propose a robust algorithm for designing such solutions. Even though there is a significant number of studies conducted on various route optimization problems in the literature, combining the constraints of depot mobility and range capacity in a VRP variation puts forth a novel research topic as well as reflecting an emerging practical challenge. In this context, the contributions of this paper are as follows:

– A novel VRP variation, the Mobile Depot VRP (MoDVRP) is introduced to reflect an emerging practical challenge in the operations field;
– MoDVRP is extended with a target coverage (number of targets visited) maximization objective function to address range constraint, which is termed Capacitated MoDVRP (C-MoDVRP); and
– A Genetic Algorithm (GA-CMoD) is proposed to solve C-MoDVRP and problems with similar structure.

The following sections of this paper are arranged in the following way. Section 1 presents the related work. Section 2 provides the problem definition. Section 3 explains our proposed algorithm on adapting the genetic operators and local search methods. Section 4 presents the results of computational tests on the performance of the proposed solution along with the alternatives. Finally, Sect. 5 concludes the study.

2 Related work

A very comprehensive VRP taxonomy was presented by Eksioglu et al. (2009). Besides revealing some exclusive statistical details concerning the VRP literature, their findings also clearly point out the rising trend of VRP research since its emergence. A complementary literature review to this one was conducted by De Jaegere et al. (2014) regarding the period 2009–2013. Both of these taxonomies categorize the VRP literature by five different aspects, as study type (theory, methodology, etc.), scenario characteristics (load splitting constraint, service time, etc.), problem characteristics (number of vehicles, vehicle homogeneity, etc.), information characteristics (evolution of information, quality of information, etc.) and data characteristics (real world, synthetic, etc.). They further break down these categories, thus providing a well-structured portrait of the work conducted in the field.

In another study, Pillac et al. (2013) classify VRP in the form of four combinations of two aspects; these are dynamic vs. static and deterministic vs. stochastic inputs. These combinations cover the possibilities whether the customers to be visited are defined or changing during the travel of the vehicle, and to what extend the change can be anticipated beforehand.

The most basic form of a VRP can be considered as a direct descendant of TSP (Lawler et al. 1985), where there is one salesman and one fixed depot, the rules are to visit all customers once and only once, and ending the route at the depot it started. In default definition, the travel distance between any two customers is equal in both directions, which implicitly makes a VRP problem a symmetric one. If reversing a travel direction changes the travel distance, this problem is defined as asymmetric VRP (A-VRP) (Vigo 1996).

The availability of more than one vehicle enables the customers to be shared among these vehicles. This variation of the VRP is named Multiple VRP (mVRP) (Taillard et al. 1996). The problem further varies with certain constraints, such as the individual range capacities of the vehicles and/or service time requests of the customers.

The problem type where multiple depots are available to serve the customers is named Multiple-Depot VRP (MDVRP) (Tillman 1969). In practice, distribution of goods such as petroleum, pharmaceutical drugs, cigarettes, etc., usually fits well for this model. If there is no capacity constraint on vehicles, then the multiplicity of the depots is interpreted as the alternative to end the route in, in which case the problem characteristic turns to be partially open ended. Otherwise, the vehicles have to include a depot in their itinerary to reset their capacity as needed before continuing their travel.

The problems where each customer needs to be served within a given time interval are classified as VRP with Time Windows (VRP-TW) (Bräysy and Gendreau 2005).

In real-world scenarios, vehicles usually have to consider certain constraints such as range and load capacities. The problems where this fact is taken into consideration are classified as Capacitated VRP (CVRP) (Kek et al. 2008). Such problems may require solutions to minimize the number of vehicles required to visit the customers. Another expectation from a solution for this problem could be designing routes, via which more customers can be visited. As the goal is to maximize the number of targets to be covered by a UAV, our study addresses this variation of the VRP in part. Previous studies related to target maximization include (Ergezer and Leblebicioglu 2013; Karakaya 2014b; Sevinç and Karakaya 2015; Karakaya 2014a); yet these studies consider only fixed depot.

The concept of depot mobility is an emerging constraint introduced as a consequence of the current trends in the operations field. This constraint presents a limited degree of resemblance to the dynamic VRP (DVRP) (Pillac et al. 2013), where customer requirements such as demand, service times and request times may be updated any time during the execution of the task from the perspective of unpredictable updates of information. In DVRP, updated information concerns only the remaining part of the task and, hence, has a limited scope of effect; whereas in the mobile depot VRP (abbreviated as MoDVRP henceforth), an update in the itinerary recursively affects the whole design. In an MoDVRP, inclusion/exclusion of a customer (in this case, target) triggers a series of reciprocal changes: the change in the tour length causes the location of the mobile depot to change. A change in the depot location alters the selection of targets to be included in the tour, and a change in the selected targets causes the tour length to change again. In this context, the Truck-and-Trailer routing problem (TTRP) (Chao 2002) also has some similarity to the MoDVRP. In TTRP, there are two types of customers, one of which can be selected as a local depot, from where the vehicle can plan service to the other type of customers. One practical example of this problem is urban delivery project of the TNT Company (Verlinde et al. 2014). The solution to this problem can be perceived as solving a series of VRPs coupled with each other in a specific way. Contrary to the MoDVRP, this problem has a finite number of local depots to select whose locations are still fixed during the execution of the local mission. In Savuran and Karakaya (2015), we study MoDVRP without considering the range constraint. In this work, we consider both range constraint and depot mobility (to term C-MoDVRP), and propose a novel solution. To the best of our knowledge, the VRP studies discussed in this paragraph come closest in similarity to C-MoDVRP and the concept of depot mobility has not been subject to other VRP studies in the literature.

3 Problem definition

VRP is defined as a combinatorial optimization problem, where the goal is to design optimal routes for one or more vehicles to visit n number of customers dispersed on a geographical area, which was first proposed by Dantzig and Ramser (1959). VRP has many variations depending on different constraints (Pisinger and Ropke 2007) and many practical problems, especially in logistics and operations fields, can be modeled with one of these variations.

In most of the practical applications of this problem, for example for a postman, salesperson or truck, all customers have to be visited, and most of the time, it is not an option to skip a node defined in their task list. Also, these assets have fixed depots, where they start and end their travels. However, considering the conditions of a carrier-deployed UAV, range constraint may force the UAV to quit the mission before visiting all targets. What is more, it has to return to a depot (in this case, the carrier) that keeps on changing locations while the UAV is in the air. To address such issues, we extend the VRP as C-MoDVRP, by introducing the constraints of depot mobility and range capacity, to conform with the following conditions:

– The UAV is deployed on a mobile platform that keeps on moving on its own route during the execution of the UAVs' mission;
– The UAV has to return to the carrier within its given flight range; and
– The goal is to visit as many targets as possible within these constraints.

4 Proposed algorithm

There are both exact and heuristic-based methods proposed for solving VRP variations. The exact methods which study VRP include Spanning Tree and Shortest Path Relaxation (Christofides et al. 1981), Branch-and-Cut, Branch-and-Cut-and-Price (Ropke 2005), and Branch-and-Bound (Laporte 1992). Comprehensive surveys of exact methods can be seen in Toth and Vigo (2002) and Baldacci et al. (2012). As the solution space of a VRP grows exponentially with the number of nodes it contains (e.g., city, target, etc.), the inclusion or exclusion of a customer in a problem may change the size of the solution space exponentially. Therefore, a VRP is defined as a Non-deterministic Polynomial (NP) complete problem (Papadimitriou 1977). This property causes exact algorithms quickly run out of computational resources. For this reason, this area of research has mainly been a subject of heuristic and meta-heuristic search methods. Exact methods can only solve moderate size VRPs to optimality (Ropke 2005). Some of the heuristic-based methods applied on VRP include Tabu Search (Amberg et al. 2000), Ant Systems (Bullnheimer et al. 1999), Artificial Bee Colony (Szeto et al. 2011), Particle Swarm Optimization (Goksal et al. 2013), Simulated Annealing (Chiang and Russell 1996), and Genetic Algorithm (Ambati et al. 1991; Larranaga 1999). Surveys on heuristic-based methods can be found in Kilby et al. (2000) and Tan et al. (2001).

In this study, we propose a genetic algorithm (GA), called Genetic Algorithm for Mobile Depot Problem (GA-CMoD), to solve the Capacitated Mobile Depot Vehicle Routing Problem (C-MoDVRP). As discussed above, meta-heuristics are efficient methods to handle combinatorial problems and, therefore, they are employed to solve variants of VRP more than any other solution techniques (De Jaegere et al. 2014). GA is one of the matured meta-heuristic optimization techniques which have been frequently and effectively applied for a variety of problems from different disciplines (Goren et al. 2010). Among other meta-heuristic optimization methods that are known to be capable of reaching near optimal or best known solutions for the same problem sets, we opt for GA because of its matured methodology and proven robustness. GA is an evolution-inspired meta-heuristic search method developed by John Holland in the 1960s (Mitchell 1998) and adopts the basic Darwinian principle of *Survival of the fittest*. GA can be applied to a large variety of problems as long as their possible solutions meet the basic requirements of being encoded as a combination of their building blocks and having an objective function. In a combinatorial problem, search space is defined as all possible combinations of its elements. GA takes a random set of these combinations for sampling, then cyclically applies genetic operations such as crossover, mutation and selection on them. Each of these cycles is called a *generation*. A generic GA flow is presented in Fig. 1.

For the representation of a generic VRP in GA, each of the nodes to be visited is taken as a *gene*, the solution strings that the combinations of these genes form are represented as *chromosomes*, and the collection of these chromosomes in every generation is called a *population*. The objective function of a generic VRP problem is the calculation of the total length of a tour. The fitness value of each chromosome is obtained by applying the objective function on them. A selection operation determines which chromosomes to survive in each generation depending on their fitness value. The theory of GA assumes that, as a result of the stochastic manipulations on individuals and selection of better fit ones in every generation, the fitness of the population will gradually grow and converge to optimality.

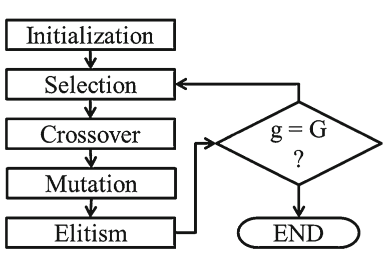

Fig. 1 Flow chart for a generic GA

In the proposed GA-CMoD solution, we have built our evolution strategy to evolve the population for greater target coverage and adapted the genetic operators to facilitate this. In our approach, similar to the encoding mentioned above, each target in the problem corresponds to a gene and any tour that a combination of a set of these targets form is a chromosome. In a generic VRP, vehicles are assumed to have unlimited ranges and, therefore, they can visit all the customers. As a result of this assumption, in a generic VRP, any possible solution has the same number of genes with the number of existing customers. However, in C-MoDVRP, vehicle has a range limitation and the number of customers visited in a tour can vary according to the created tour. To simulate the dynamic nature of a solution with different number of visited customers, in our implementation, a chromosome does not have to contain all of the genes (targets), conforming to the fact that a range constrained tour may not cover all of the targets. Also, a tour in our algorithm includes the takeoff and landing points besides the targets to be visited. In the following, the methodology applied in GA-CMoD is introduced.

4.1 Assumptions

For simplicity, we assume that the carrier and the UAV move at constant speeds and the carrier moves on a given constant heading, starting from a given geographical point in the area. However, as long as a route function for the carrier is provided, our approach can be applied on more sophisticated carrier routes as well.

4.2 Inclusion of takeoff and landing points in the solution

Determination of takeoff and prediction of landing points are critical in creating an efficient and feasible tour. GA-CMoD employs some geometrical calculations to acquire the nearest takeoff and landing points during the course of the carriers' move. For additional explanations of these calculations see "Appendix".

4.3 Initialization

The role of this operation in a GA is to take a sample of the search space as the initial population. Some popular methods for initialization are generating random chromosomes, performing multiple random initializations to choose the best one, and using heuristic methods to feed in a better initial population (Coley 2010). GA-CMoD generates random chromosomes by adding random targets to a tour as long as the total tour length is kept within range. To make sure the range constraint is not violated, the distance to the takeoff and landing points are included in the tour length calculation.

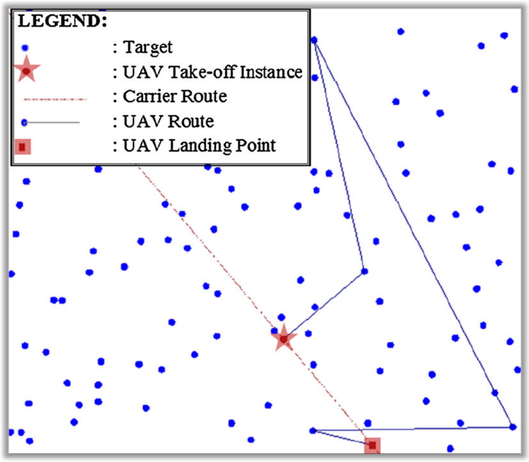

Fig. 2 A randomly generated chromosome

Figure 2 represents a sample of a chromosome generated in this manner.

Chromosomes consisting of such tours are generated randomly up to the defined number of population (N) to form the initial population.

4.4 Crossover

The function of a crossover operation in a GA is to bring together partial information blocks from different solution candidates to form new ones. Donor and receiver solutions involved in this information exchange process are, respectively, named as *parent* and *offspring* chromosomes in GA terminology. When combined with the selection operation as in a natural GA flow, this process works in a way to bring together the better individual traits from the parents into the offspring.

As the purpose in the defined C-MoDVRP is maximizing the targets to be covered within a given range, in our case the distance-wise density of the targets in a sub-set of a chromosome can be considered a better trait. As such, GA-CMoD seeks to bring together better sub-sets from different parents. For this, we customize the merge crossover operation studied previously by Pereira et al. (2002). In GA-CMoD implementation, merge crossover operation joins the sub-sets of two chromosomes at the geographically nearest target nodes they contain. This operation selects two nodes from different chromosomes such that they are closer to each other than one of them is to its adjacent node. On the sample given in Fig. 3, if we assume node 20 of the second parent is geographically the nearest one to node 9 of the first parent, provided that node 2 is not closer to the node 9 than the node 20, they will produce the offspring as shown in the figure.

The resulting chromosomes may lack feasibility as there might be repetitive genes in their structure and their tour

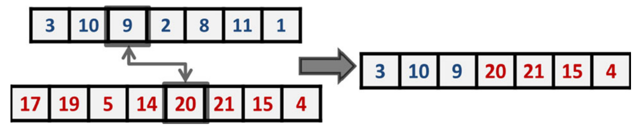

Fig. 3 Merge crossover operation sample

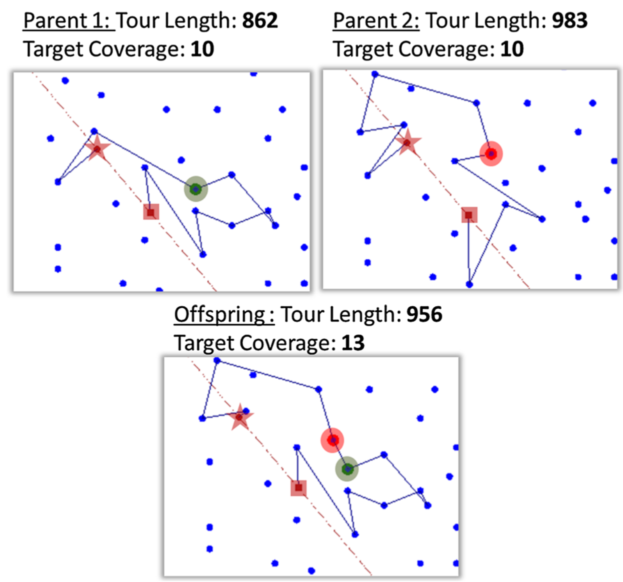

Fig. 4 A sample product of merge crossover operation

length might exceed the range. Therefore, an extra process of removing repetitive genes is employed. Another process to fit the chromosome into range trims it at a random end. A sample end product of the *Merge crossover operation* is shown in Fig. 4.

4.5 Mutation

Mutation operation in a GA helps to preserve genetic diversity and its purpose is to refrain from local optima and allow a more thorough exploration of the search space. In GA-CMoD, we employ an exchange mutation operation in a heuristic way that replaces an included remote target in a chromosome with a non-included closer one. For this, depending on the probability, from a random set of included targets, the one that has the greatest total distance of its both edges is removed and from non-included targets, the one that has the smallest total distance to both of its adjacent targets at (a random) insertion point is inserted, with the condition that the tour length not exceeding the range. Figure 5 presents an example chromosome mutated in this way.

4.6 Insertion local search

To further improve the candidate solutions evolved during the course of the GA-CMoD, we employ insertion local search

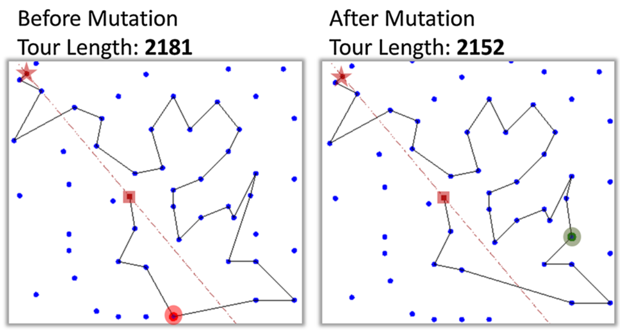

Fig. 5 A sample product of heuristic exchange mutation operation

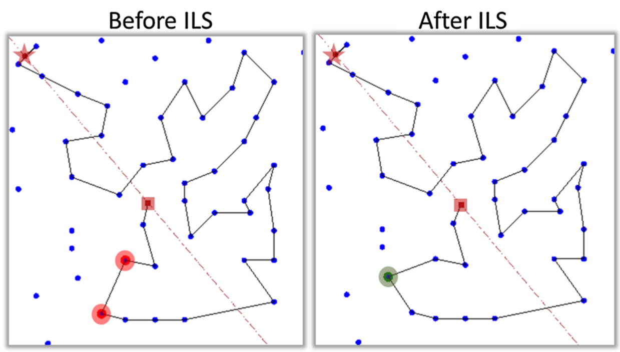

Fig. 6 A sample product of ILS method

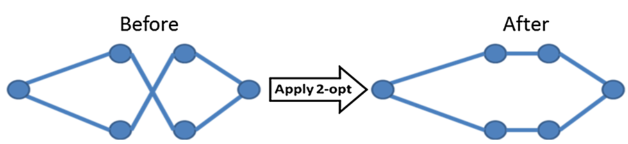

Fig. 7 Illustration of a 2-opt operation (reproduced from Karakaya 2015)

(ILS) heuristic on a given probability. Among the different variations of this local search method such as nearest, cheapest, farthest and random (Hoos and Stützle 2004), the nearest insertion local search fits well for our case. In this method, depending on the probability, the algorithm picks a random target that is included in the chromosome and inserts the nearest non-included target next to it as long as the insertion does not violate the range constraint. Figure 6 represents an insertion operation taking place during a run of the algorithm.

4.7 2-opt local search

2-opt is a local search method (Croes 1958) that modifies a route to remove intersections and is commonly employed in TSP-based problems (Englert et al. 2007). To accomplish this, it iteratively reverses one of the two adjacent sub-tours in a tour and, if the new tour is shorter than the original one, it is rearranged with this order as in the sample in Fig. 7.

Due to its heavy computational workload, this operation is applied only on the fittest chromosome of each generation. To adopt it for C-MoDVRP, GA-CMoD updates the takeoff and landing locations of a tour when it is rearranged by the 2-opt.

4.8 Fitness

The fitness value of a chromosome determines its probability to survive and reproduce in the next generations (Mitchell 1998). In GA-CMoD, survivability of a chromosome is determined by one of the two different criteria: the number of the targets it covers or the geographical density of them. Where the purpose of the first criteria obviously is to satisfy the objective of the problem, the second one helps selecting the chromosomes that has higher tour productivity. Tour productivity of a chromosome is formulated as follows: $f = C^2/L$ where f is fitness, C is the number of targets covered and L is the tour length.

4.9 Elitism

The stochastic nature of the genetic operations may occasionally draw the evolution backward too, in the sense that when a chromosome in one generation is modified by an operator, it may get less fit in the next generation. Elitism operation is applied to prevent the potential drawback through generations by cloning the most fit chromosome in every generation and preserving this copy from genetic operations (Mitchell 1998). We apply elitism on the chromosome that has the greatest target coverage.

4.10 Selection

Selection operation is the process where the principle of survival of the fittest is put in place in a GA. For this, the fitness value of each individual is calculated using the objective function of the problem and the better fit individuals are selected over the less fit ones to remain in the next generation. There are many sophisticated selection methods (Mitchell 1998). We implement the fitness proportionate (also known as the roulette wheel) selection. This method sets the survival probability of a chromosome proportionate to its fitness value; therefore, it not only makes a distinction between the bad and the good, but also between the good and the better (Coley 2010).

GA-CMoD employs a Hybrid Selection approach by composing the two fitness functions explained above. In this approach, one half of the population is selected by the number of targets they cover, and the other half by their tour productivity (i.e., geographical density of targets in a tour). The goal of this selection is to give a chance of survival to the chromosomes that may have higher tour productivity but lower target coverage. Our expectation is that, regardless of their individual target coverage, such chromosomes may provide valuable genetic material for the whole population by means of crossover operation that will follow.

5 Computational results

The combining of the operators explained in the previous section as in Fig. 8 forms the GA-CMoD.

To monitor the performance of our proposed GA-CMoD algorithm, we both test the effects of its operators (genetic operators and local search methods) individually by comparing them with rival operators and the overall performance of the algorithm with Nearest Neighbor and Hill Climbing heuristics. In the following, we describe our test approach and present the test results.

5.1 Simulation environment

The simulation tests are conducted using 16 of benchmark problems from Heidelberg TSP Library (of Heidelberg 1995). Selection of these problems was determined by the consideration of sampling a fairly wide scope of problem sizes within the limitations of the computational power available during the tests. For each problem, number of generations (G) and population size (N) parameters are tuned with sensitivity tests. Unless otherwise stated, all tests were repeated ten times for each problem throughout our experiments, and their averages are presented. The details of the simulation environment are explained below.

5.1.1 Characteristics of selected benchmark problems

Each selected TSP problem is tested with three different UAV ranges: short range (SR), medium range (MR), and long range (LR). The best known solutions in the literature (of Heidelberg 1995) for the selected TSP problems are used

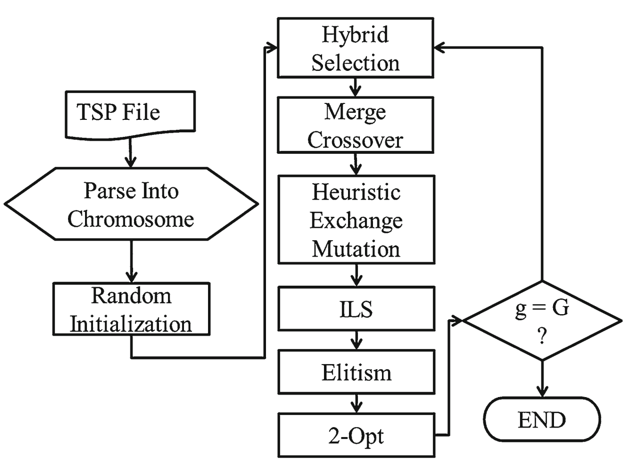

Fig. 8 Flow chart for GA-CMod

Table 1 Characteristics of selected benchmark problems

Problem and no. of nodes	Known optimal TL	SR 25 %	MR 50 %	LR 75 %
Att48	10,628	2657	5314	7971
Berlin52	7542	1885	3771	5656
Eil76	538	134	269	403
Pr76	108,159	27,039	54,079	81,119
KroB100	22,141	5535	11,070	16,605
Eil101	629	157	314	471
Lin105	14,379	3594	7189	10,784
Bier127	118,282	29,570	59,141	88,711
Ch130	6110	1527	3055	4582
Pr144	58,537	14,634	29,268	43,902
Ch150	6528	1632	3264	4896
Rat195	2323	580	1161	1742
KroB200	29,437	7359	14,718	22,077
A280	2579	644	1289	1934
LinHP318	41,345	10,336	20,672	31,008
Att532	27, 686	6921	13,843	20,764

Table 2 Fixed parameters used in GA-CMoD

Crossover rate (P_c)	Mutation rate (P_m)	ILS rate (P_{ils})
0.60	0.02	0.02

to decide these ranges by setting their values proportionate to the known best tour length of the given problem as seen in Table 1. For example, the best known tour length for the Att48 TSP problem is 10628. For short range (SR), we use 25 % of the known best tour length which is 2657. Similarly, we use 50 % of the known best tour length (5314) as medium range (MR) and 75 % of the known best tour length (7971) as long range (LR).

5.1.2 Tuning the GA parameters

Characteristics of a C-MoDVRP problem such as number and topology of nodes and actual values of the given constraints require optimized GA parameters for a better performance. To determine the GA parameters to be used throughout the tests, we have conducted sensitivity tests on each TSP problem as follows. Rates of crossover, mutation, and insertion local search methods are set fixed to the values typically used in the literature (Mitchell 1998; Coley 2010) as in Table 2.

Then, different population size (N) and number of generations (G) values are combined together in a single solution setup and the results of these combinations are evaluated.

As an example result, Fig. 9 presents percentage of covered targets for the Eil101 problem when medium range (MR) constraint is considered and GA-CMoD is run with

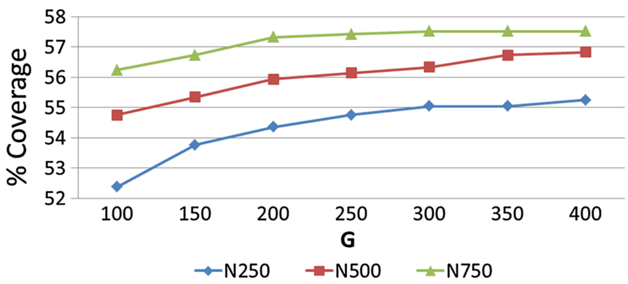

Fig. 9 Performance of N and G parameters for Eil101

Table 3 Determined N and G parameters in GA-CMoD for each test case

Problem	Pop. size (N)	Generations (G)		
		SR	MR	LR
Att48	750	80	100	120
Berlin52	500	120	150	200
Eil76	250	150	200	300
Pr76	750	150	200	300
KroB100	750	150	200	300
Eil101	750	150	200	300
Lin105	500	150	200	300
Bier127	500	200	300	400
Ch130	250	200	300	400
Pr144	750	200	300	400
Ch150	750	200	300	400
Rat195	750	250	350	450
KroB200	750	250	350	450
A280	250	500	600	700
LinHP318	750	550	650	800
Att532	750	700	900	1100

different number of population sizes and generations. Percentage of targets covered (TC) is formulated as

$$\mathrm{TC} = \frac{\mathrm{VT}}{\mathrm{TT}} \times 100 \tag{1}$$

where VT is the number of visited targets in a tour and TT is the number of total targets that benchmark problem contains. Evaluating these results, the N and G parameter set for this particular instance are selected as 750 and 200. Similar tests were conducted for each selected benchmark problem for three defined ranges, with the same set of population sizes and various numbers of generations ranging from 20 to 1200, depending on the problem size. Depending on these results, the GA parameter values for each problem were determined as in Table 3.

During these experiments, we observed the convergence time to be less than 30 seconds for the TSP benchmark problems with the node number less than 300 and with short range setting. However, the observed convergence times for the considered problems increase as the problem size (target number) and UAV range increase. For instance, the largest problem tested in our study with a node number 532 converged to the reported result in less than 35 min.

Table 4 Problem parameters

UAV speed	Carrier speed	Carrier start point	Carrier heading
300	40	{0,0}	45°

Table 5 Tested operators

Operator	GA-CMoD	Rival operator
Selection	Hybrid	By-coverage
Crossover	Merge	PMX, OX1
Mutation	Heuristic exchange	Displacement, insertion
Local search	ILS	None
Elitism	Elitism	Elitism
Local search	2-opt	None

5.1.3 Problem parameters

Problem parameters that were used for the problems are presented in Table 4.

As indicated in Table 4, in the test setup the carrier starts from the northwest corner of a topology and crosses diagonally towards the southeast corner with a constant speed and heading.

5.2 Individual performance of operators

To evaluate the performance of the operators employed in GA-CMoD, we replace them in GA-CMoD with some rival operators from literature and monitor the difference in their productions. For this, for each rival operator in Table 5 an alternative algorithm is built in a way that the only difference of this algorithm from GA-CMoD is that operator. Then the results produced by GA-CMoD are compared against the results produced by these algorithms in which the tested operators are replaced.

5.2.1 Performance of selection method

As explained above, the idea behind the proposed hybrid selection method is to devise a metric to associate tour productivity in the evolution process, while also satisfying the target maximization objective function. We test the performance of hybrid selection method by comparing it with by-coverage selection method, which determines the fitness of a chromosome solely depending on the number of targets it

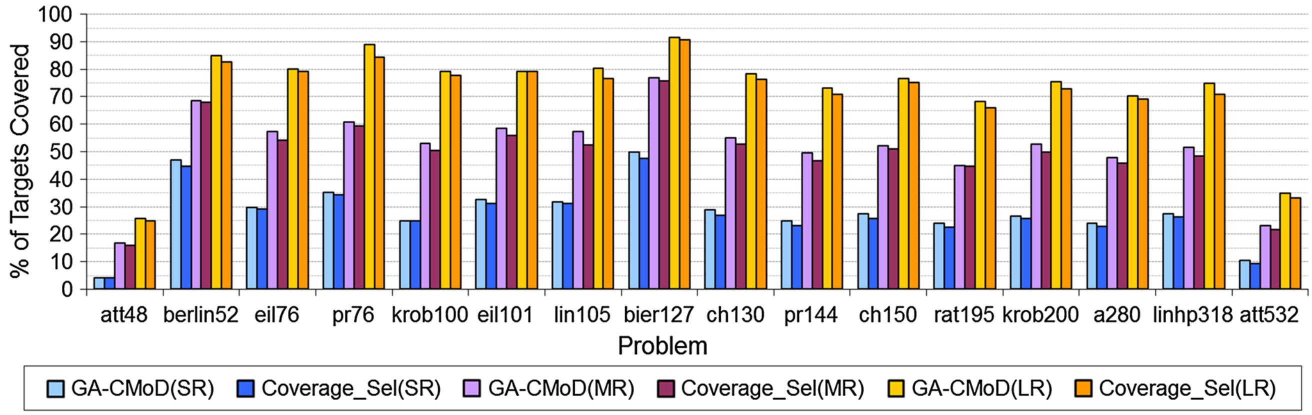

Fig. 10 Comparison of target coverage for GA-CMoD and algorithm with by-coverage selection method

covers. As such, while both hybrid and by-coverage selection methods are designed to work in favor of target maximization, hybrid selection method does this by also watching for tour productivity. Therefore, this comparison enables us to assess the effectiveness of the core idea of our proposed selection method in isolation.

The results of the comparison tests for the selection methods are given in Fig. 10. In this graph, for 16 benchmark problems and 3 different range levels the results are presented. For all the comparison graphs throughout this paper, the same figure layout is used.

Most of the times, GA-CMoD with hybrid selection method outperforms the alternative algorithm with the by-coverage selection method by covering more targets up to 11 %. In the case of the Lin105 problem with medium range setting, for example, GA-CMoD with hybrid selection covers 57.2 % of the total 105 targets, where the algorithm that employs the by-coverage selection covers 52.4 % of them. Figure 10 depicts the comparisons of target coverage by GA-CMoD and the by-coverage selection algorithms in percentage of the number of total targets the problems contain.

Our expectation from hybrid selection method is, while primarily driving the evolution towards greater target coverage, also manipulating the population to reproduce more efficiently. The comparison results show that, as seeking for both target coverage and tour-productivity, hybrid selection method enables GA-CMoD to perform superior over the algorithm with by-coverage selection method which focuses on target coverage only. In 46 out of 48 test cases, GA-CMoD performs superior over the algorithm with by-coverage selection, by producing solutions with more target coverage varying from 0.8 to 11.5 %. Only in one test case, the algorithm with by-coverage selection produces better solution than that of GA-CMoD with about 0.4 % better coverage. In the case of Att48(SR) they both consistently produce the same results, which can be explained with the very limited scope of probabilities that can be produced within the constraints of this particular instance.

5.2.2 Performance of crossover operator

The purpose of this test is to evaluate the performance of merge crossover operator, which was tailored specifically for the constraints of the C-MoDVRP, over the ordinary crossover operators from the literature. The effectiveness of the proposed merge crossover operator is tested by replacing it with the PMX and OX1 rival crossover operators. These operators are among the most popular crossover operators in the TSP literature (Larranaga 1999). They both adopt the principle of exchanging some random portions between two parents and they differ on how they repair the offspring in case of infeasibility.

As observed in the results, GA-CMoD with merge crossover consistently outperforms the alternative one with PMX (see Fig. 11) by covering more targets from 0.1 to 48.1 %, and the one with OX1 (see Fig. 12) by covering more targets from 1.3 to 31.2 %. GA-CMoD produces better solutions than both the alternative algorithms for every test case, except for the case of Att48(SR) where the results are the same. It can be inferred from the results that reproduction of generations through the logic of merge crossover can prove more effective for the optimization of a C-MoDVRP.

5.2.3 Performance of mutation operator

Concentrating the targets in a tour as much as possible is a key property of GA-CMoD. For contributing to achieve this goal while also supporting genetic diversity within the population, the proposed heuristic exchange mutation operator implements an exclusive logic that involves vertex distance calculations in determining the targets to be included in a tour. To test the performance of this operator we replace it with Insertion and Displacement mutation operators. These both are popular operators for mutation in TSP literature (Larranaga 1999), designed to swap portions within a tour using their own logic. With this test, we aim to measure the success of heuristic exchange mutation in GA-CMoD, by comparing

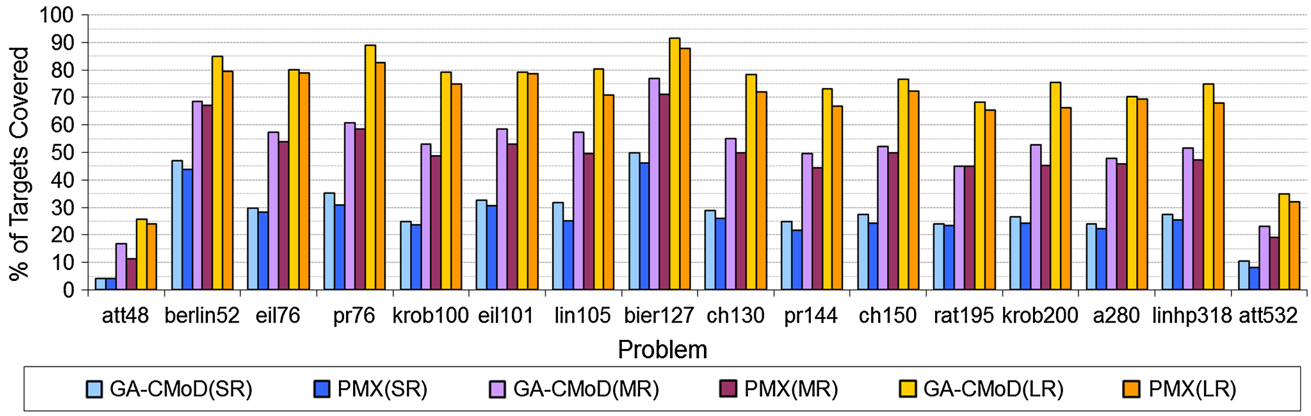

Fig. 11 Comparison of target coverage for GA-CMoD and algorithm with PMX crossover operator

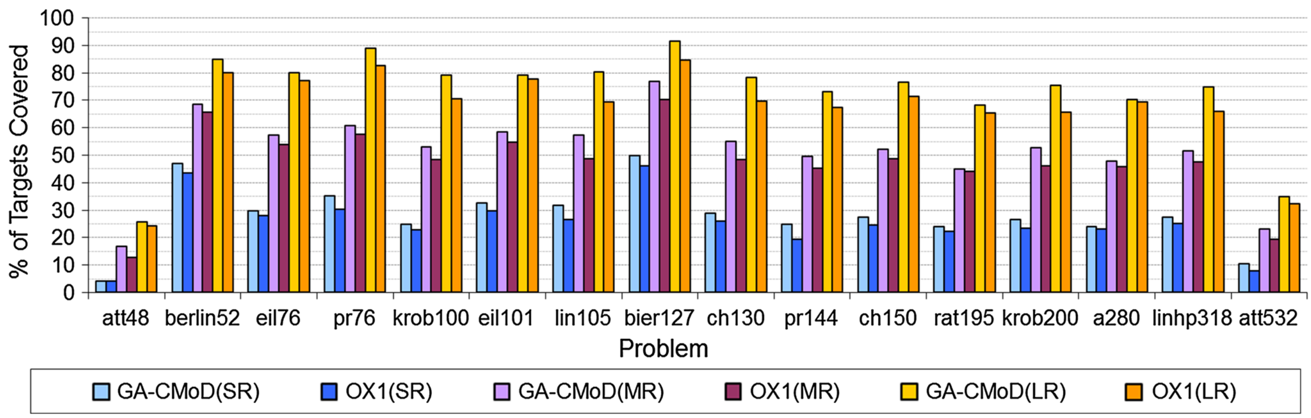

Fig. 12 Comparison of target coverage for GA-CMoD and algorithm with OX1 crossover operator

it with existing mutation operators in the literature whose only task is to preserve genetic diversity.

According to the test results, GA-CMoD with heuristic exchange mutation consistently outperforms the alternative algorithm with the Displacement mutation operator (see Fig. 13) by covering more targets from 1.6 to 29.4 %, and the other algorithm that employs the Insertion mutation operator (see Fig. 14) by covering more targets from 3.1 to 24 %. For every test case except for Att48(SR) again, GA-CMoD produces superior results.

5.2.4 *Performance of insertion local search method*

GA-CMoD implements two local search methods for fine tuning of the converging solutions. Among these, Insertion Local Search (ILS) works on increasing target coverage, while 2-opt is responsible for improving the tour length, both within the context of local optimization.

To investigate the effects of ILS, we compare its existence against its nonexistence in GA-CMoD.

In 46 out of 48 test cases GA-CMoD outperforms the one that does not employ ILS (see Fig. 15) by covering more targets from 0.2 to 23 %. In only one test case the algorithm with ILS produces a better result by 0.7 % more coverage. As read from the results, the ILS method has proven to be an effective local optimization method for a C-MoDVRP.

5.2.5 *Performance of 2-opt local search method*

The other employed local search method in GA-CMoD, 2-opt, rearranges a tour for targets to be visited more efficiently. This rearrangement in return may create room for further expansion of the tour over new targets. Therefore, to examine the impacts of this method, we remove it from GA-CMoD for comparison, and expect to see a difference of convergence.

As observed in the test results, GA-CMoD consistently outperforms the algorithm that does not employ 2-opt local search method in 47 out of 48 test cases, with covering more targets varying from 2.1 to 50.8 % (see Fig. 16). 2-opt already is a reliable local search method in VRP research, with these results, it is safe to argue that once adapted properly, it can also serve for C-MoDVRP.

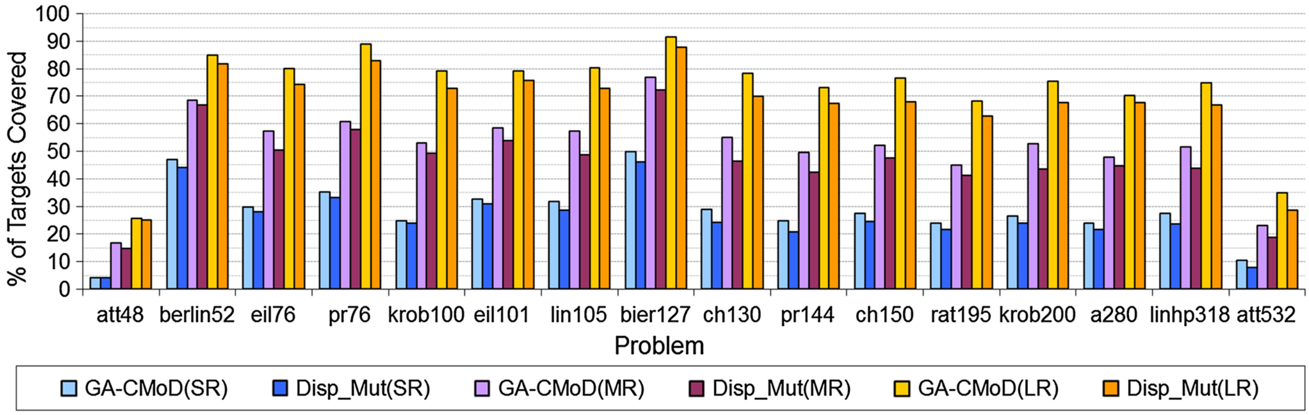

Fig. 13 Comparison of target coverage for GA-CMoD and algorithm with displacement mutation operator

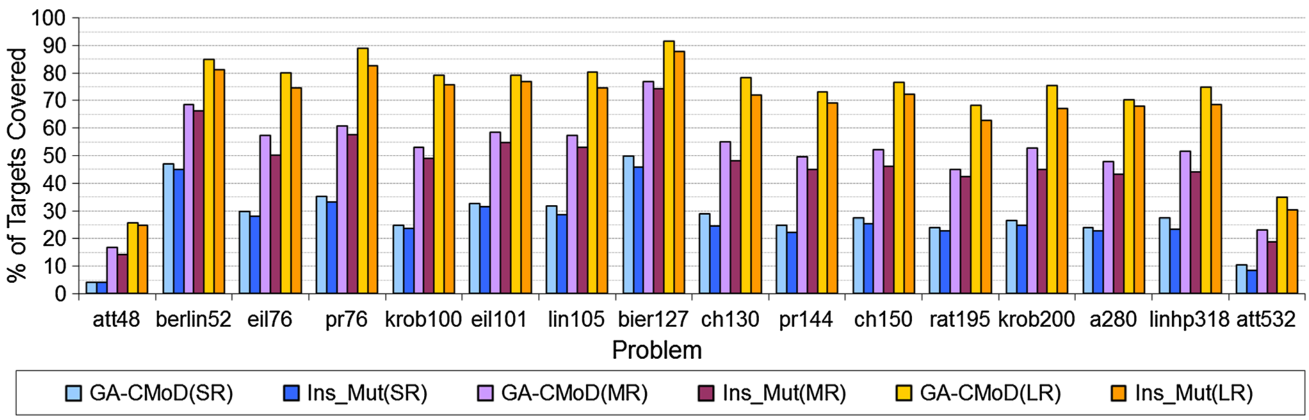

Fig. 14 Comparison of target coverage for GA-CMoD and algorithm with insertion mutation operator

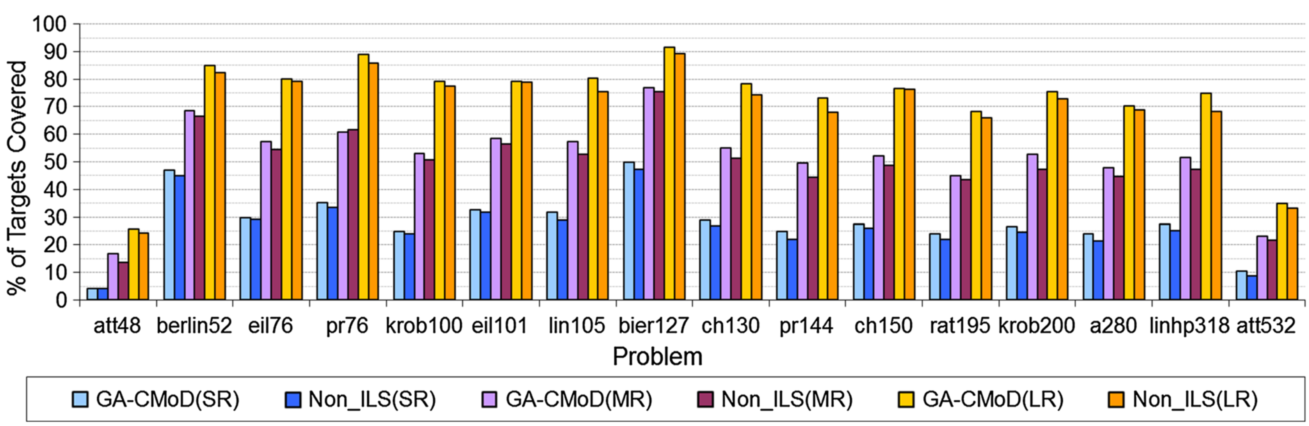

Fig. 15 Comparison of target coverage for GA-CMoD with and without ILS method

5.2.6 Summary

So far, we have investigated the individual performances of the operators employed in GA-CMoD. As seen from the results, genetic operators and local search methods that were specifically tailored for the constraints of a C-MoDVRP have shown their success over standard GA operators. In the below section, we compare the proposed GA-CMoD solution with two well-known heuristics.

5.3 Overall performance evaluation

For testing the performance of GA-CMoD, we compare its performance with the performance of the Nearest Neigh-

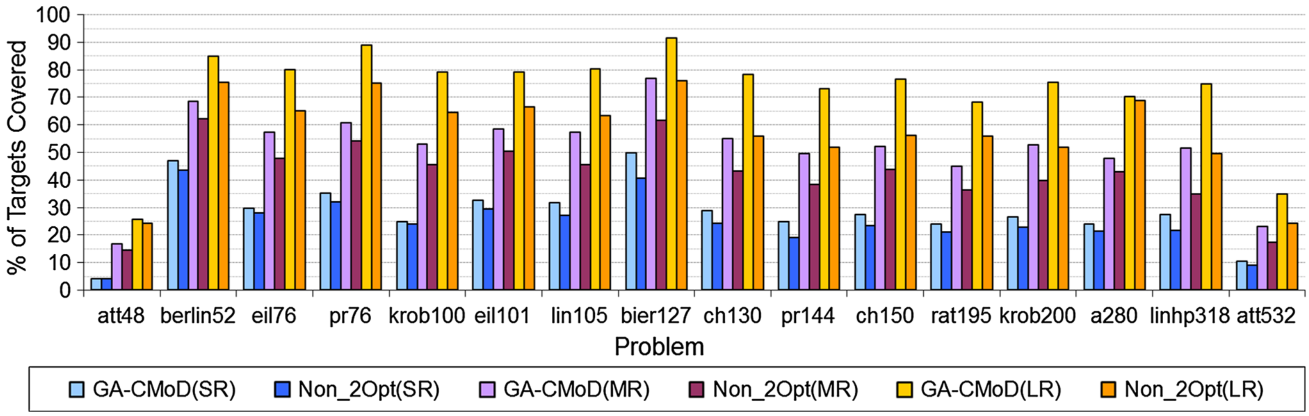

Fig. 16 Comparison of target coverage for 2-opt and non-2-opt methods

bor (NN) and Hill Climbing (HC) Heuristics. Both of these heuristics were reinforced with the 2-opt local search method for better performance.

5.3.1 Comparison with the nearest neighbor heuristic

Nearest Neighbor is a well-known constructive search algorithm that is one of the earliest methods proposed for TSP problems adopting the principle of selecting the next nearest unvisited node until all nodes have been covered (Lawler et al. 1985). It runs fast, the optimality of the tours it produces, however, highly depends on the layout of the nodes given in the problem instance.

To improve performance of NN, we enhance it with the 2-opt local search. This approach was examined thoroughly by Maniezzo et al. (2008). We adopt this enhancement as follows. The carrier route is partitioned into takeoff steps in such a frequency that selecting any other takeoff point between two adjacent takeoff steps would cause no change in the route design, thus eliminating the possible disadvantage of selecting superfluous number of takeoff steps. Then, starting from each of these takeoff steps, a range-fitting tour is designed by the NN method. Once the range constraint forces the tour to complete, it is rearranged with 2-opt and if this arrangement brings out sufficient room, new targets are added from the last point on, using the NN approach again. This process is repeated until the tour cannot be expanded over new targets anymore. As an example, Fig. 17 demonstrates the difference between the solutions produced by NN (only) and NN and 2-opt combination, for Ch130(SR) problem.

As a result, among the tours designed for each takeoff step in this way, the one that has the greatest target coverage is selected as the output of NN and 2-opt method (referred to as *NN* henceforth). Figure 18 compares the best results produced by both NN and GA-CMoD for all three ranges.

According to these results, in 42 out of 48 test cases, GA-CMoD performs superior to NN, by covering more targets from 1.8 to 75.6 %. In 13 test cases, GA-CMoD covers

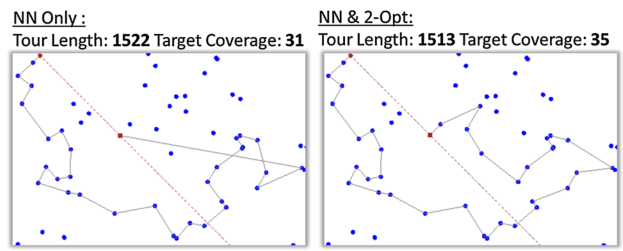

Fig. 17 Comparison of solutions produced by NN-only and NN and 2-opt for Ch130 (SR)

more than 10 % more targets compare to NN. The problem Rat195, however, presents an exception, which can be explained by the topology-dependent nature of the NN algorithm. As shown in Fig. 19, the pattern the nodes are laid out in this problem, coinciding with the carriers move properties, provides a natural advantage for NN method. The results can be perceived as, especially when augmented with 2-opt, NN method can perform moderate on a C-MoDVRP depending on the topology. GA-CMoD on the other hand proves to be a robust method on C-ModVRP problems of different constraints and topologies.

5.3.2 Comparison with the hill climbing heuristic

Hill climbing (HC) (Michalewicz and Fogel 2013) is another heuristic search algorithm that builds a solution by starting from a random initial solution and changing one element of it in every iteration for improvement. Since this algorithm operates only on one state of a solution at a time, it can easily encounter a local optima. This occurs when no immediate improvement can be achieved in the adjacent state of a solution. This problem can be alleviated by introducing random restart in the algorithm. In this approach, exploration is restarted with different initial states, for a certain number of times, and the best result is selected. The variation of this algorithm where the best solution is sought in every itera-

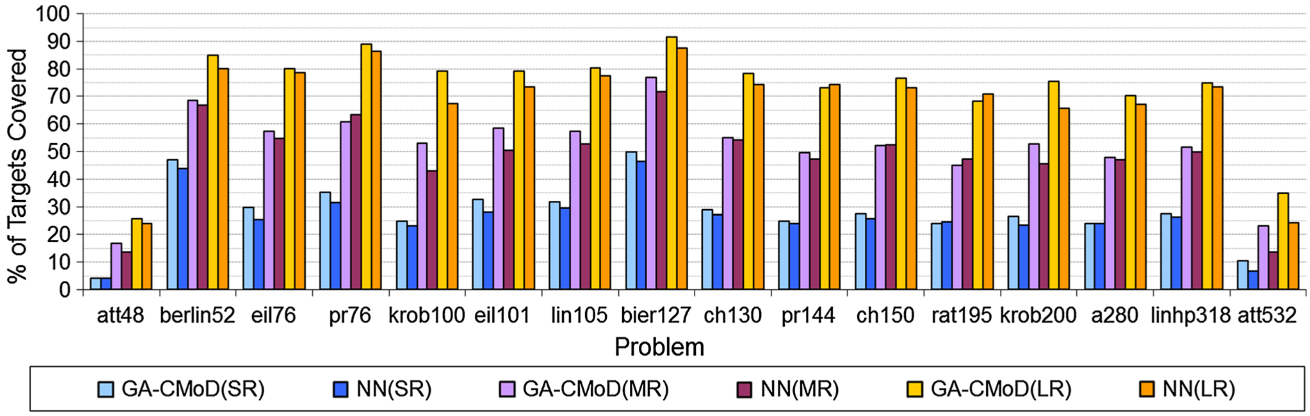

Fig. 18 Comparison of target coverage for GA-CMoD and NN heuristic

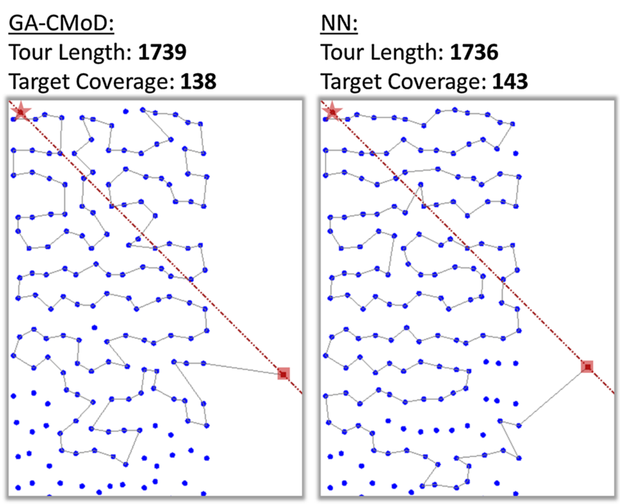

Fig. 19 Comparison of routes produced by GA-CMoD and NN heuristic for Rat195(LR)

tion instead of a better one is called the Steepest Ascend Hill Climbing (Michalewicz and Fogel 2013). In this study, we employ Steepest-Ascend Random-Restart Hill Climbing to solve C-MoDVRP in the following way. The carrier route is partitioned into takeoff steps as explained in the previous paragraph. All targets are ordered randomly in a tour without considering the range constraint, and the part of this tour that fits within the range is marked. Then starting from the takeoff point, next target is swapped with the one among the subsequent targets, which best satisfies the improvement criteria. Improvement criteria are primarily set as increasing the number of targets fitting in range constraint and if this cannot be achieved, decreasing the tour length alternatively. Once the tour cannot be improved with this process, it is rearranged with 2-opt and the hill climbing algorithm is run on from the last target in the current itinerary. This combination of Hill Climbing and 2-opt is repeated until the tour cannot be improved anymore. The best output of 10 random restarts of this process is selected as the end product of HC heuristic. The best results of both GA-CMoD and HC solutions are compared in Fig. 20. It is observed that GA-CMoD overwhelmingly outperforms HC (see Fig. 20) by covering more targets from 16.6 to 260.4 %, in all test cases with the exception of Att48 where the same result is produced. In 36 out of 48 test cases, GA-CMoD covers more than 50 % more targets compare to HC. Greater size and range of a problem means

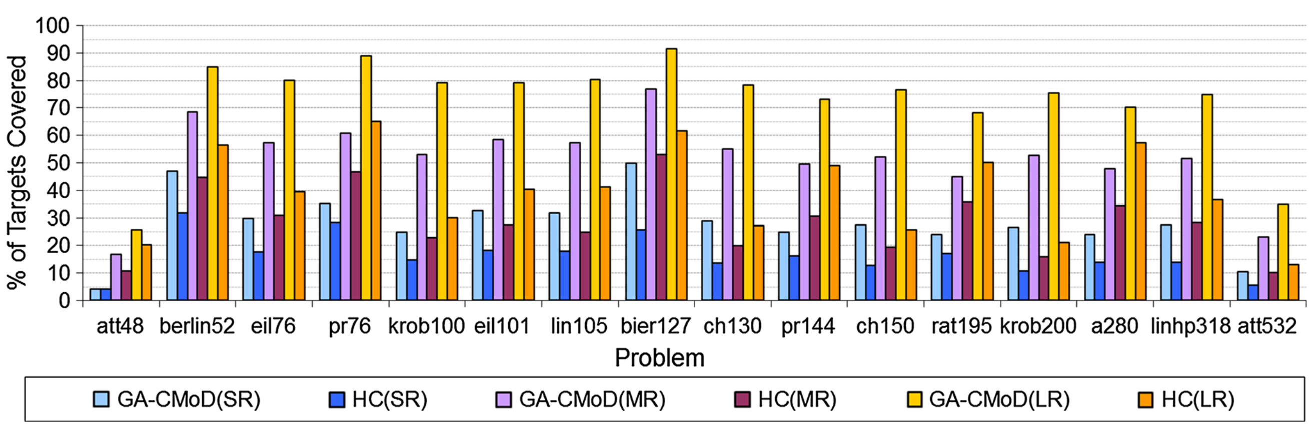

Fig. 20 Comparison of target coverage for GA-CMoD and HC heuristic

greater search space. From this perspective, GA-CMoD also tends to present more supremacy over HC as the search space gets greater, which is an indication of effectiveness. These results can be read as HC has some evident disadvantages over GA-CMoD for C-MoDVRP.

6 Conclusions

In this study, we propose novel variants of VRP in which depot mobility and range capacity constraints are introduced. We term these problems Mobile Depot VRP (MoDVRP) and Capacitated Mobile Depot VRP (C-MoDVRP). As a sample case of C-MoDVRP, we design a routing problem in which a UAV is assumed to be deployed on a mobile carrier. The generated route should cover all or most of the targets while considering UAV's flight range and the mobile carriers route. For solving this C-MoDVRP, we propose a genetic algorithm, GA-CMoD, in which the genetic operators, local search methods and the objective function of GA are designed so that the evolution proceeds towards maximizing the target coverage while respecting all constraints of the C-MoDVRP. We consider the problem constraints covered in this study correspond with some practical challenges emerging with the current trend in UAV usage.

GA-CMoD is employed to solve 16 benchmark problems for 3 different ranges and it produces better results, varying from 11 to 21 % over designed GAs and from 75 to 260 % over selected heuristics. Consistent success of the proposed algorithm over other possible solutions in 48 different scenarios is an indication of its robustness.

As for future work, we intend to introduce the constraint of multiple VRPs into this problem, where there will be multiple UAVs deployed on a mobile carrier and the objective will be to cover more targets with less total tour length by UAVs. In this scenario, the UAVs will be assigned individual target lists and takeoff and landing points during the course of the carriers move, and will try to produce maximum total efficiency of target coverage and tour length, thereby tackling with a more complex level of optimization challenge.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Appendix: Inclusion of takeoff and landing points in the solution

A.1 Takeoff point calculation

Whenever the first target to be visited in a tour is changed by an operation, the takeoff point for that tour is re-assigned with the calculations explained in this section.

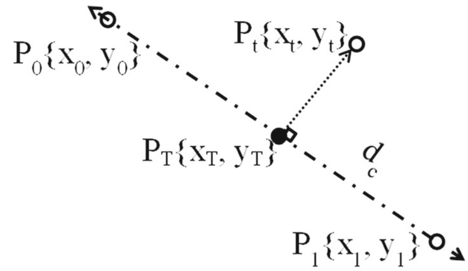

Fig. 21 Geometrical representation of nearest takeoff point calculation

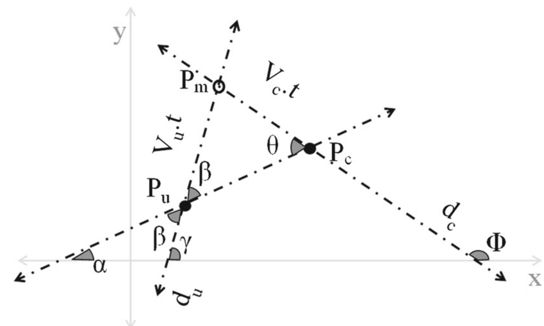

Fig. 22 Geometrical representation of nearest landing point prediction

Since the carrier moves on a constant heading, the algorithm calculates the nearest takeoff locations for each tour depending on the first target to be visited in its itinerary. For this, a linear equation of point-slope form is used to calculate the shortest path between the target and the carrier route as depicted in Fig. 21. Here P_t represents the location of the first target in the given tour, P_0 and P_1 represent any two points belonging to the line of the carrier route (d_c), and P_T represents the nearest takeoff location.

A.2 Landing point prediction

For this task, the time that the last target visited by the UAV in a tour is taken as the start point (t_0) for calculation. Since the speeds of the carrier and UAV are constant, their movement axes are vectorized from this point on and a linear equation of point-slope form is used to calculate their nearest meeting, as shown in Fig. 22. Here P_u and P_c, respectively, represent the locations of the carrier and the UAV, and P_m is the nearest meeting point.

The time spent by the UAV to visit targets can be acquired using its speed and the distance it traveled before time (t_0), using the formula $t = d/v$. Then this time value and the speed of the carrier can be used in the same formula to acquire the distance it covered at the time of (t_0). The inclusion of this distance in the formulation gives the exact location of UAV's landing on the carrier.

References

Ambati BK, Ambati J, Mokhtar MM (1991) Heuristic combinatorial optimization by simulated darwinian evolution: a polynomial time algorithm for the traveling salesman problem. Biol Cybern 65(1):31–35

Amberg A, Domschke W, Voß S (2000) Multiple center capacitated arc routing problems: a tabu search algorithm using capacitated trees. Eur J Oper Res 124(2):360–376

Arjomandi M, Agostino S, Mammone M, Nelson M, Zhou T (2006) Classification of unmanned aerial vehicles. The University of Adelaide Australia [En linea] Disponible en. http://personal.mecheng.adelaide.edu.au/maziar.arjomandi/Aeronautical20

Baldacci R, Mingozzi A, Roberti R (2012) Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints. Eur J Oper Res 218(1):1–6

Bräysy O, Gendreau M (2005) Vehicle routing problem with time windows, part I: route construction and local search algorithms. Transp Sci 39(1):104–118

Bullnheimer B, Hartl RF, Strauss C (1999) An improved ant system algorithm for thevehicle routing problem. Ann Oper Res 89:319–328

Chao IM (2002) A tabu search method for the truck and trailer routing problem. Comput Oper Res 29(1):33–51

Chiang WC, Russell RA (1996) Simulated annealing metaheuristics for the vehicle routing problem with time windows. Ann Oper Res 63(1):3–27

Christofides N, Mingozzi A, Toth P (1981) Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations. Math Program 20(1):255–282

Coley DA (2010) An introduction to genetic algorithms for scientists and engineers. World Scientific, Singapore

Croes G (1958) A method for solving traveling-salesman problems. Oper Res 6(6):791–812

Dantzig GB, Ramser JH (1959) The truck dispatching problem. Manag Sci 6(1):80–91

De Jaegere N, Defraeye M, Van Nieuwenhuyse I (2014) The vehicle routing problem: state of the art classification and review. FEB Research Report KBI_1415

Eksioglu B, Vural AV, Reisman A (2009) The vehicle routing problem: a taxonomic review. Comput Ind Eng 57(4):1472–1483. doi:10.1016/j.cie.2009.05.009

Englert M, Röglin H, Vöcking B (2007) Worst case and probabilistic analysis of the 2-opt algorithm for the tsp. In: Proceedings of the eighteenth annual ACM-SIAM symposium on discrete algorithms. Society for Industrial and Applied Mathematics, pp 1295–1304

Ergezer H, Leblebicioglu K (2013) Path planning for uavs for maximum information collection. IEEE Trans Aerospace Electron Syst 49(1):502–520

Goksal FP, Karaoglan I, Altiparmak F (2013) A hybrid discrete particle swarm optimization for vehicle routing problem with simultaneous pickup and delivery. Comput Ind Eng 65(1):39–53

Goren HG, Tunali S, Jans R (2010) A review of applications of genetic algorithms in lot sizing. J Intell Manuf 21(4):575–590

Heidelberg U (1995) Tsplib. http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/XML-TSPLIB/instances/

Hoos HH, Stützle T (2004) Stochastic local search: foundations and applications. Elsevier, Amsterdam

Karakaya M (2014a) A local optimization technique for assigning new targets to the planned routes of unmanned aerial vehicles. Balkan J Electr Comput Eng 2(2)

Karakaya M (2014b) Uav route planning for maximum target coverage. Comput Sci Eng (CSEIJ) 4(1)

Karakaya M (2015) Msct: an efficient data collection heuristic for wireless sensor networks with limited sensor memory capacity. KSII Trans Internet Inf Syst 9(9):3396–3411

Kek AG, Cheu RL, Meng Q (2008) Distance-constrained capacitated vehicle routing problems with flexible assignment of start and end depots. Math Comput Model 47(1):140–152

Kilby P, Prosser P, Shaw P (2000) A comparison of traditional and constraint-based heuristic methods on vehicle routing problems with side constraints. Constraints 5(4):389–414

Laporte G (1992) The vehicle routing problem: an overview of exact and approximate algorithms. Eur J Oper Res 59(3):345–358

Larranaga P (1999) Genetic algorithms for the traveling salesman problem: a review of representations and operators. Artif Intell Rev 13(2):129–170

Lawler EL, Lenstra JK, Kan RA, Shmoys DB (1985) The traveling salesman problem: a guided tour of combinatorial optimization. Wiley, New York

Maniezzo V, Battiti R, Watson JP (2008) Learning and intelligent optimization: second international conference, LION 2007 II, Trento. Selected Papers, vol 5313. Springer

Martin A, Dewolfe RA (2012) Promising outlook for navys unmanned aviation. National Defense

Michalewicz Z, Fogel DB (2013) How to solve it: modern heuristics. Springer Science & Business Media

Mitchell M (1998) An introduction to genetic algorithms. MIT Press, Massachusetts

Nonami K (2007) Prospect and recent research and development for civil use autonomous unmanned aircraft as uav and mav. J Syst Design Dyn 1:120–128

Papadimitriou CH (1977) The euclidean travelling salesman problem is np-complete. Theor Comput Sci 4(3):237–244

Pearson I et al (2006) The way ahead for maritime uavs. Tech. rep, DTIC Document

Pereira FB, Tavares J, Machado P, Costa E (2002) Gvr: a new genetic representation for the vehicle routing problem. In: Artificial intelligence and cognitive science. Springer, Berlin, pp 95–102

Pillac V, Gendreau M, Guéret C, Medaglia AL (2013) A review of dynamic vehicle routing problems. Eur J Oper Res 225(1):1–11

Pisinger D, Ropke S (2007) A general heuristic for vehicle routing problems. Comput Oper Res 34(8):2403–2435

Ropke S (2005) Heuristic and exact algorithms for vehicle routing problems. Unpublished PhD thesis, Computer Science Department, University of Copenhagen

Savuran H, Karakaya M (2015) Route optimization method for unmanned air vehicle launched from a carrier. Lecture Notes Softw Eng 3(4):279–284

Sevinç E, Karakaya M (2015) Maximizing uav target coverage under flight range and target service time constraints. Lecture Notes Softw Eng 3(4)

Sullivan MJ, Masters T, Greifner L, Hadley J, Hassinger K, Jezewski L, Lea M, Pendleton J, Persons TM, Sun R (2013) Defense acquisitions: navy strategy for unmanned carrier-based aircraft system defers key oversight mechanisms. Tech. rep, DTIC Document

Szeto W, Wu Y, Ho SC (2011) An artificial bee colony algorithm for the capacitated vehicle routing problem. Eur J Oper Res 215(1):126–135

Taillard ED, Laporte G, Gendreau M (1996) Vehicle routeing with multiple use of vehicles. J Oper Res Soc, 1065–1070

Tan KC, Lee LH, Zhu Q, Ou K (2001) Heuristic methods for vehicle routing problem with time windows. Artif Intell Eng 15(3):281–295

Tillman FA (1969) The multiple terminal delivery problem with probabilistic demands. Transp Sci 3(3):192–204

Toth P, Vigo D (2002) The vehicle routing problem, SIAM Monographs on Discrete Mathematics and Applications. SIAM, Philadelphia

Verlinde S, Macharis C, Milan L, Kin B (2014) Does a mobile depot make urban deliveries faster, more sustainable and more economically viable: results of a pilot test in brussels. Transp Res Proc 4:361–373

Vigo D (1996) A heuristic algorithm for the asymmetric capacitated vehicle routing problem. Eur J Oper Res 89(1):108–126

Watts AC, Ambrosia VG, Hinkley EA (2012) Unmanned aircraft systems in remote sensing and scientific research: classification and considerations of use. Remote Sens 4(6):1671–1692