Noname manuscript No.
(will be inserted by the editor)

An external archive guided NSGA-II algorithm for Multi-Depot Green Vehicle Routing Problem

Meriem HEMICI · Djaafar ZOUACHE · Brahmi BOUALEM · Kaouther HEMICI

Received: date / Accepted: date

Abstract The paper proposes a new elitist non-dominated sorting genetic algorithm II, called External Archive Guided Elitist Non-dominated Sorting Genetic Algorithm II (EAG-NSGA-II) to solve the multi-depot green vehicle routing problem (MDGVRP). In the proposed algorithm, the elitist non-dominated sorting genetic algorithm II (NSGA-II) is incorporated with adaptive local search to improve the solution′s quality and greatly accelerate the convergence. To ensure a good balance between convergence and diversity, we employ the external archive based on adaptive epsilon dominance. The experiments are conducted over famous 11 Cordeau′s data sets and compared against three well-known algorithms, including Strength Pareto evolutionary algorithm (SPEA2), Elitist non-dominated sorting genetic algorithm (NSGA-II), and Multi-objective evolutionary algorithm based on decomposition (MOEA/D). The results indicate that the proposed algorithm is highly competitive and outperforms the selected state-of-the-art multiobjective optimization algorithms.

M. HEMICI
Department of Mathematics, University of Mohamed El Bachir El Ibrahimi, Bordj Bou Arreridj; Algeria
Tel.: +213-66-9712518
E-mail: hemicimeriem20@yahoo.com

D. ZOUACHE
Department of Computer Science, University of Mohamed El Bachir El Ibrahimi, Bordj Bou Arreridj; Algeria

B. BOUALEM
Department of Operation research, University of Mohamed El Bachir El Ibrahimi, Bordj Bou Arreridj; Algeria

K. HEMICI
Department of Economic Sciences, Commerciales and Management Sciences, University of El-Chahid Hama Lakhdar, El-oued; Algeria

Keywords Multi-Depot Green Vehicle Routing Problem · Multiobjective optimization · Evolutionary algorithms · Epsilon dominance · Crowding distance · Genetic algorithm

1 Introduction

Green Supply Chain Management (SCM) aims to integrate environmental thinking into supply chain management. Over the years, Green supply chain management (GrSCM) has gained increasing interest among researchers and practitioners of operations and supply chain management.
The green vehicle routing problem (GVRP) is one of the most important problems in green supply chain management, this problem generates a set of routes with a set of the customer which consumes a determined demand corresponds to the quantity of product to deliver. A fleet of vehicles identical capacity vehicles is available at the depot to satisfy the demands of customers. A vehicle starts from a depot, serves customers one-by-one, and, ends its trip in the same depot. Instead of the single-depot problem, the multi-depot problem is proposed in the multi-depot green vehicle routing problem (MDGVRP), it is more complicated than other variants of the vehicle routing problems.
In the literature, there are many papers are addressed the green multi-depot vehicle routing problem. Erdoğan and Miller-Hooks [2] used a mixed-integer-linear programming (MILP) formulation and two heuristics. The first heuristic is a modified Clarke and Wright savings algorithm (MCWS) and the second heuristic is a density-based clustering algorithm (DBCA). Zhang et al [5] proposed a Two-stage Ant Colony System (TSACS) that uses two distinct types of ants for two different purposes. The first type of ant is used to assign customers to depots, while the second type of ant is used to find the routes.
In this paper, we propose a new variant of elitist non-dominated sorting genetic algorithm II, called External Archive Guided Elitist Non-dominated Sorting Genetic Algorithm II (EAG-NSGA-II), to solve MDGVRP with two main contributions. The first one is the inclusion of an adaptive local search to greatly accelerate the convergence. The second is the employ of the external archive based on adaptive epsilon dominance to ensure a good balance between convergence and diversity. Epsilon dominance impacts the size of the archive; as ϵ increases, the size of the archive decreases.
The experimental results show that the proposed algorithm has a better performance compared to the selected state-of-the-art multiobjective optimization algorithms on famous 11 Cordeau$'$s data sets.
The rest of the paper is structured as follows. Section. 2 presents the multi-objective formulation of the considered MDGVRP. In Section. 3, EAG-NSGA-II is proposed and illustrated. Computational Results in Section. 4. Finally, we conclude this work and address some open problems in Section. 5.

2 Multi-depot green vehicle routing problem

The MDGVRP can be defined as follows. Let $G = (V, A)$ be a directed graph, with $V = N \bigcup D$ is the vertex set, and A is the arc set. Vertex set $N = \{1, 2, .., n\}$ represents the customers to be served, and set $D = \{1, 2, ..., m\}$ represents the depots. Each Vertex $i \in V$ is geographically located at coordinates (x, y). The arc set A denotes all possible connections between the vertices V. Each customer $i \in N$ has a demand of goods $q_i > 0$. The travel costs or distance is associated with each arc $(i, j) \in A$ from $i \in N$ to $j \in N$,i.e. $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$. Also, set M consists of homogeneous vehicles with capacity limit Q. To serve all customers in N, the MDGVRP constraints should be satisfied as follows:

(1) each vehicle starts and ends the route at the same depot.
(2) every customer vertex is visited on exactly one vehicle.
(3) the total load of vehicle k does not exceed Q.

Notations:
N**:** The number of customers.
M**:** The number of depots.
K**:** The number of vehicles.
Q**:** The capacity of vehicle.
d_{ij}**:** The traveling distance between customers i and j.
CCF**:** Carbon emission conversion factor.
Decision variable:

$$x_{ij}^k = \begin{cases} 1 & If\ the\ vehicle\ \mathrm{h}\ of\ depot\ \mathrm{d}\ visits\ customer\ \mathrm{j}\ from\ customer\ \mathrm{i} \\ 0 & Otherwise \end{cases}$$

The mathematical model for the GMDVRP is given as follows:

$$min\ \ F_1 = \sum_{k=1}^{M} \sum_{i=1}^{N} \sum_{j=1}^{N} d_{ij} x_{ij}^{dk} \times CCF \tag{1}$$

$$min\ \ F_2 = \sum_{k=1}^{M} \sum_{i=1}^{N} x_{0i}^{dk} \tag{2}$$

$$\sum_{k=1}^{M} \sum_{i=1}^{N} x_{ij}^{dk} = 1 \qquad \forall i \in \{1, ..., N\} \tag{3}$$

$$\sum_{i=1}^{N} x_{0i}^{dk} = 1 \qquad \forall k \in \{1, ..., M\} \tag{4}$$

$$\sum_{i=1}^{N} x_{ih}^{dk} - \sum_{j=1}^{N} x_{hj}^{dk} = 0 \qquad \forall h \in \{1, ..., N\}, \forall k \in \{1, ..., M\} \tag{5}$$

$$\sum_{i=1}^{N} x_{iN+1}^{dk} = 1 \qquad \forall k \in \{1, ..., M\} \tag{6}$$

$$\sum_{i=1}^{N} q_i \sum_{j=1}^{N} x_{ij}^{dk} \preceq Q \qquad \forall k \in \{1, ..., M\} \tag{7}$$

In this formulation, the objective function (1, 2) refers to minimizing the total carbon emissions produced by all the vehicles and the number of vehicles respectively. The equations (3) state that each customer must be visited once by one vehicle,equations (4, 5 and 6) ensure that each vehicle leaves and returns to the depot. The equations (7) state that the vehicle capacity should not be exceededs.

3 The proposed algorithm (EAG-NSGA-II) for MDGVRP

In this section, we present a new variant of elitist non-dominated sorting genetic algorithm II, called External Archive Guided Elitist Non-dominated Sorting Genetic Algorithm II (EAG-NSGA-II) for solving the MDGVRP. Firstly, an initial population P_0 of $\mid P_0 \mid$ feasible solutions is generated randomly. Then, two objectives functions are evaluated for each solution in the initial population. The first objective represents the total travel distances as given in equation (1) and the second objective is the number of vehicles as given in equation (2), where they must be minimized simultaneously, subject to constraints mentioned above (see section 2). Our approach employs another external population called archive containing the nondominated solutions of the initial population of solutions. The solutions are then subjected to an iterative evolutionary process until the termination condition is met. The evolutionary process is performed by genetic operators, which are the tournament selection operator and the crossover operator called Best Cost Route Crossover (BCRC) to generate offspring solutions Q. The offspring solutions $Q^{'}$ are improved by adaptive local search. The offspring solutions $Q^{'}$ are used to update the external archive. Finally, a combination of the population and offspring population is performed to obtain a population $R = P \bigcup Q$ that is sorted according to the principle of dominance and classified into different fronts (F_i, i = 1, 2, ..., etc.) where all non-dominated solutions of the population are assigned the first front F_1, then they are removed from the population. All non-dominated solutions of the population are assigned the second front F_2, then they are removed from the population. And so on. This process is iterated until the entire population is sorted. The reader is referred to [1] for additional details about NSGA-II. The current population of size N is filled with the two steps is:
Step 1: Create a current population $P_{current} = \oslash$. Set $i = 1$. While $\mid P_{current} \mid + \mid F_i \mid < N$, do $P_{current} = P_{current} \bigcup F_i$ and $i = i + 1$.
Step 2: Sort the front F_i according to the crowding distances and include the $(N- \mid P_{current} \mid)$ solutions with the largest distance values in the population

$P_{current}$.
The proposed algorithm EAG-NSGA-II for the MDGVRP is illustrated in Algorithm 1.

Algorithm 1 Pseudocode of EAG-NSGA-II algorithm for MDGVRP

Step I: Initialization
1: Generate an initial population P_0 of $\mid P_0 \mid$ feasible solution.
2: Evaluate the initial population P_0 with the two objective functions.
3: Initialize the external archive A by the non-dominated solutions set of the initial population.

Step II: New solution generation
1: **Selection:** Select parents Q from the population P by using tournament selection.
2: **Reproduction:** Apply crossover operators on Q to generate Q^*.
3: **Local search:** Apply adaptive local search to improve Q^*.
4: **Replacement:**
  – $R = Combination(P, Q^*)$.
  – Sort the non-dominated solution of $R = \{F_1, F_2, F_3, ...\}$.
  – Create a current population $P_{current} = \oslash$, and i=1.
  while $\mid P_{current} \mid + \mid F_i \mid < N$
    - $P_{current} = P_{current} \bigcup F_i$.
    - i=i+1.
  endwhile
  – Sort the front F_i according to the crowding distances.
  – $P_{current} = P_{current} \bigcup F_i(1 : N- \mid P_{current} \mid)$.
5: **Update:**
  Update of external archive by Q^*.

Step III: Stopping phase
1: If termination condition are satisfied, then output the external archive.

3.1 Solution representation

The solution is represented by three basic steps (Scheduling, Grouping, Routing), like that seen in Figure 1.

3.2 Crossover operator

The crossover is an important operation in a genetic algorithm. Moreover, the crossover operators are responsible for producing the offspring for the selected parents with probability p_c so as to maintain the desired characteristics of the selected parents. The crossover (Best Cost Route Crossover (BCRC) [8] is used to minimizing the total carbon emissions produced by all the vehicles and the number of vehicles simultaneously by respecting the feasibility constraints. In

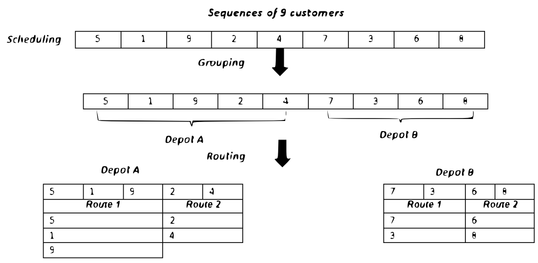


Fig. 1 Example of solution presentation.

practice, the Best Cost Route Crossover (BCRC) can be summarized as follows:
Step 1: Choose two parents P_1, P_2 from the tournament selection.
Step 2: Randomly select a depot from each parent and select a route from each parent under that depot in a random manner.
Step 3: Remove all customers belonging to route 1 from parent 1 and remove all customers belonging to route 2 from parent 2. Step 4: Reinsert the removed customers from each parent to form the offspring. For every customer belonging to route 1 (or route 2).
- Compute the cost of insertion of route 1 (or route 2) into each location of parent 2 (or parent 1) and store the costs in an ordered list.
- For each insertion location, check whether the insertion is feasible or not.
- If there are no possible insertion locations in the unremoved route, then a new route was created.

3.3 Adaptive local search

The role of adaptive local search (ALS) is vital in EAG-NSGA-II in order to greatly accelerate the convergence. Two types of neighborhood structures are used in local search to further improve offspring solutions, which are interdepot and intradepot neighborhoods. The roulette wheel selection is used to decide which neighborhood method is applied to improve the select solution.
Interdepot neighborhood methods: These are constructed based on several basic operators, including swap, shift, 2-opt$*$, Or-opt, and 2-opt.
Intradepot neighborhood methods: These are constructed based on several basic operators, including a cross, large neighborhood search (LNS), inversion, and 2-opt$*$.

3.3.1 Updating the archive

In this work, we use an external archive based on the adaptive parameter of ϵ. Initially, the external archive contains all the nondominated solutions of the initial population, since EAG-NSGA-II is an iterative algorithm, this external

archive should be updated in each iteration; so, we employ Epsilon dominance approach to update the external archive′s solutions.
For any archive solution A and offspring solution c, we associate an identification vector ($box = (box_1, box_2, ..., box_M)^T$, where M is the total number of objectives) as follows:

$$box_j(f) = \lfloor\ (f_j)/\log(1+\epsilon)\rfloor \tag{8}$$

Where $\lfloor.\rfloor$ is the absolute value, f_j: the objective value jth of an archive solution and ϵ denotes the admissible error. The identification vector divides the whole objective space into hyper-box. An offspring solution c is compared with all the archive solutions A according to Epsilon dominance concept to decide if this solution is accepted into the archive as shown in the algorithm 2. More precisely, the EAG-NSGA-II algorithm compares offspring solution with all the archive solutions according to three conditions [9]:

1. If the identification vector of the offspring solution dominates the identification vector of any archive solution, the offspring solution is accepted and the archive solution is deleted.
2. On the other hand, if the identification vector of the offspring solution is dominated by the identification vector of any archive solution, then it means that the offspring solution is Epsilon dominated by this archive solution and so the offspring is rejected.
3. If neither of the above two cases occurs, then it means that the offspring is equivalent with the archive solutions in terms of Epsilon dominance relation. We separate this third case into two conditions:
   3-1 If the offspring solution and an archive solution a are shared the same identification vector, so we have to verify if the offspring solution dominates the archive solution, or the offspring solution is non-dominated to the archive solution but is closer to the identification vector than the archive solution, then the offspring solution is accepted.
   3-2 In the event of an offspring solution not sharing the same identification vector with any archive solution, the offspring solution is accepted.

This means that the archive maintains the diversity by allowing only one solution to be present in each hyper-box on the Pareto front.

4 Computational Results

The experimental setup is outlined in this section, the results are presented, and the discussion is provided. The proposed algorithm is implemented in MATLAB on an Intel (R) Core (TM) i3 - 6006UCPU2:0GHz PC with a Windows 10 to 64 - bit operating system. The results presented below are based on the following set of EAG-NSGA-II parameters:

– Size of the population: 50.
– Maximum number of generation: 500.

Algorithm 2 Pseudocode of Update the external archive procedure

Require: A: The external archive; f: The offspring solution;
Calculate the identification vector for f and all solutions of the archive A;
$D := \{\forall a \in A \mid box_f < box_a\}$
  if $D \neq \oslash$ **then**
    $A := A \cup f \setminus D$
  else
    if $\exists a : (box_a = box_f \wedge f < a)$ **then**
      $A := A \cup f \setminus \{a\}$
    else
      if $\nexists a : (box_a = box_f \vee box_a < box_f)$ **then**
        $A := A \cup f$
      else
        $A := A$
      end if
    end if
  end if

- Probability of crossover: 0.9.
- Value of ε: 0.001.
- Value of CCF: 0.2

Table.1 shows the computational results for instances used by Cordeau, et al. [3]. In presenting the Pareto-optimal set-based results, we provided the best solution (considering both the minimization of the CO2 emission and the number of vehicles) per a given problem. In some instances (like problems 1 and 3, 18, and 21), we have two or more solutions in each instance because neither one dominates the other. The boldface values in columns labeled indicate the problem instances where our algorithm outperforms the best-known results in the literature by reducing the number of vehicles or it has same the number of vehicles.
In the figure 2, we can see clearly that the convergence of the EAG-NSGA-II

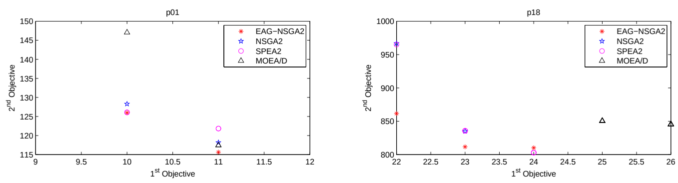
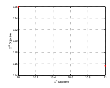
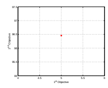
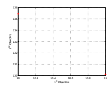
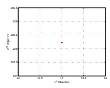
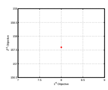
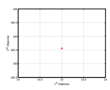
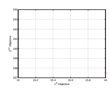
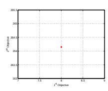
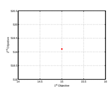
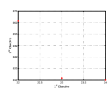
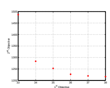


Fig. 2 Pareto set obtained by the multi-objective algorithms on some instances.

Table 1 Computational results for instances used by Cordeau, et al. [3]

Instance	N	M	Objective 1, Total number of vehicles	Objective 2, CO2 emission [kg]	Pareto's Front
p01	50	4	11 **10**	115.64 125.97	
p02	50	4	**5**	96.46	
p03	75	5	11 **10**	130.12 135.45	
p04	100	2	**15**	208.22	
p05	100	2	**8**	157.60	
p06	100	3	**15**	183.56	
p07	100	4	16 **15**	187.51 193.72	
p12	80	2	**8**	264.15	
p15	160	4	**15**	519.11	
p18	240	6	24 23 **22**	810.11 811.66 861.52	
p21	360	9	38 37 36 35 34 **33**	1216.76 1219.86 1226.96 1252.55 1283.54 1486.46	

algorithm is better than NSGA-II, SPEA-II, and MOEA/D algorithms.

5 Conclusions

This paper proposed a new variant of elitist non-dominated sorting genetic algorithm II, called External Archive Guided Elitist Non-dominated Sorting Genetic Algorithm II (EAG-NSGA-II) for solving the MDGVRP. Adaptive local search was integrated to greatly accelerate the algorithm convergence. The external archive based on adaptive epsilon dominance was employ to ensure a good balance between convergence and diversity. EAG-NSGA-II was evaluated through using the famous 11 Cordeau′s data sets. The experimental results proved that EAG-NSGA-II was effective to solve MDGVRP. EAG-NSGA-II was significantly superior over NSGA-II, SPEA-II, MOEA/D in all instances.
For future work, EAG-NSGA-II can be tested to solve other variants of the vehicle routing problems such as green multi-depot VRPTW. EAG-NSGA-II can be further developed using a novel update strategy to solve similar combinatorial optimization problems in the field of supply chain management.

References

1. Deb, Kalyanmoy and Pratap, Amrit and Agarwal, Sameer and Meyarivan, TAMT, *A fast and elitist multiobjective genetic algorithm: NSGA-II*, IEEE transactions on evolutionary computation, 6, 182–197 (2002)
2. Erdoğan, Sevgi and Miller-Hooks, Elise, *A green vehicle routing problem*, Transportation research part E: logistics and transportation review, 100–114, Elsevier, 2012.
3. Cordeau, Jean-François and Gendreau, Michel and Laporte, Gilbert, *A tabu search heuristic for periodic and multi-depot vehicle routing problems*, Networks: An International Journal, 30,2, 105–119, 1997,Wiley Online Library.
4. Renaud, Jacques and Laporte, Gilbert and Boctor, Fayez F, *A tabu search heuristic for the multi-depot vehicle routing problem*, Computers & Operations Research, Elsevier, 1996.
5. Zhang, Weiheng and Gajpal, Yuvraj and Appadoo, Srimantoorao and Wei, Qi and others, *Multi-depot green vehicle routing problem to minimize carbon emissions*, Sustainability, Multidisciplinary Digital Publishing Institute, 2020.
6. Zitzler, Eckart and Thiele, Lothar, *Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach*, IEEE transactions on Evolutionary Computation, IEEE, 1999.
7. Zitzler, Eckart and Laumanns, Marco and Bleuler, Stefan, *A tutorial on evolutionary multiobjective optimization*, Metaheuristics for multiobjective optimisation, 3–37,Springer, 2004.
8. Ombuki, Beatrice and Ross, Brian J and Hanshar, Franklin, *Multi-objective genetic algorithms for vehicle routing problem with time windows*, Applied Intelligence, 17–30, vol 24, Springer, 2006.
9. Mishra, Sudhansu Kumar and Ganapati, P and Meher, Sukadev and Majhi, Ritanjali, *A fast multiobjective evolutionary algorithm for finding wellspread pareto-optimal solutions*, In KanGAL Report No. 2003002, lndian Institute Of Technology Kanpur, Citeseer, 2002.