D-MAENS2: A Self-adaptive D-MAENS Algorithm with Better Decision Diversity

Qingquan Zhang*, Feng Wu†, Yang Tao*, Jiyuan Pei*, Jialin Liu* and Xin Yao*
*Guangdong Provincial Key Laboratory of Brain-inspired Intelligent Computation
Department of Computer Science and Engineering
Southern University of Science and Technology
Shenzhen 518055, China
Email: qingquanzhangchn@gmail.com
†Northwestern Polytechnical University
Xi'an, Shaanxi, 710129, China

***Abstract*—The capacitated arc routing problem is a challenging combinatorial optimization problem with numerous real-world applications. In recent years, several multi-objective optimization algorithms have been applied to minimize both the total cost and makespan for capacitated arc routing problems, among which the decomposition-based memetic algorithm with extended neighborhood search has shown promising results. In this paper, we propose an improved decomposition-based memetic algorithm with extended neighborhood search, called D-MAENS2, which uses a novel method to construct a gene pool to measure and improve the diversity of solutions in decision variable space. Additionally, D-MAENS2 is capable of adapting online its hyper-parameters to various problem instances. Experimental studies show that our novel D-MAENS2 significantly outperforms D-MAENS on 81 benchmark instances and shows outstanding performance on instances of large size.**

***Index Terms*—Capacitated arc routing problem, local search, memetic algorithms, meta-heuristics, multiobjective optimization.**

I. Introduction

The capacitated arc routing problem (CARP), a well-known combinatorial optimization problem, has many practical applications in the real world, such as snow removal [1], winter gritting [2] and waste collection [3]. CARP aims to figure out an optimal routing plan to serve a predefined set of tasks which locate on arcs under some constrains. Since CARP is a NP-hard problem [4], exact approaches can hardly solve large-scale CARPs in real-world scenarios within reasonable time.

In the past few decades, numerous studies indicate that meta-heuristics are effective to approximate the optimum solutions of CARPs [5]–[10]. One typical algorithm belonging to meta-heuristics is the tabu search [11], which adopts a neighborhood structure to solve CARPs. Such algorithms include the guided local search [12], the tabu scatter search [13], the memetic algorithm (MA) [14] and the MA with extended neighborhood search (MAENS) [5], arguably the most popular meta-heuristic for solving CARPs. MAENS [5] uses merge-split, a local search operator, to avoid trapping into local optima and achieves promising performance in terms of the total cost of routes in its recommended solutions.

However, solutions optimized by minimizing the total costs of routes are not always suitable in real-world scenarios as multiple criteria should be considered and balanced in real life. A typical observation in such solutions is the extremely uneven costs among different routes for vehicles. To address this problem, Lacomme et al. [15] proposed a multiobjective CARP (MO-CARP) which considers at the same time the total cost of a solution and the *makespan*, defined as the maximum cost of all routes in the solution. Mei et al. [16] further adopted the framework of MOEA/D [17] and NSGA-II [18], and proposed D-MAENS for MO-CARP. Although D-MAENS [16] showed its superior performance in solving MO-CARP benchmark instances compared against MAENS [5], it has some drawbacks. Firstly, D-MAENS [16] is configured by two hyper-parameters which are often tuned in an offline manner and then remain unchanged when dealing with different problem instances. Furthermore, from the perspective of evolutionary multiobjective optimization algorithms (MOEAs), NSGA-II [18] still has weaknesses [19] in dealing with MO-CARPs due to the special characteristics of multiobjective optimization problems (MOPs) transformed from CARPs. Diversity in decision space is also an essential metric in CARPs. On one hand, it helps to improve global search capabilities; on the other hand, decision makers expect more solutions to choose from under the similar cost to facilitate the handling of uncertain events in reality.

In this paper, we propose a novel decomposition-based memetic algorithm with extended neighborhood search, called D-MAENS2, to tackle the two issues described above. D-MAENS2 differs from D-MAENS mainly in 3 ways: (i) D-MAENS2 uses a novel metric to measure the diversity of solutions in decision variable space, which is closer to the needs in real life. Decision makers often care more about how the route plans differ from each other when their costs are

This work was supported by the National Key R&D Program of China (Grant No. 2017YFC0804003), the National Natural Science Foundation of China (Grant No. 61906083, 61976111), the Guangdong Provincial Key Laboratory (Grant No. 2020B121201001), the Program for Guangdong Introducing Innovative and Enterpreneurial Teams (Grant No. 2017ZT07X386), the Shenzhen Science and Technology Program (Grant No. KQTD2016112514355531), the Science and Technology Innovation Committee Foundation of Shenzhen (Grant No. JCYJ20190809121403553) and the Program for University Key Laboratory of Guangdong Province (Grant No. 2017KSYS008).

Corresponding author: Jialin Liu (*liujl@sustech.edu.cn*).

acceptable, rather than how diverse the cost values are. (ii) SPEA2 [20] is adopted to D-MAENS2 to maintain the diversity. (iii) Moreover, D-MAENS2 adapts its hyper-parameters in an online manner according to different optimization states when dealing with different problem instances.

The rest of this paper is organized as follows. The background of CARP and D-MAENS is introduced in Section II. Section III describes our proposed D-MAENS2. The experimental studies are presented in Section IV. Section V concludes and discusses some potential future work.

II. Background

A. Capacitated Arc Routing Problem

An undirected capacitated arc routing problem (CARP) [4], [21] is defined as follows. Let $G = (V, E)$ denote a connected undirected graph. $V = \{0, \ldots, n\}$ is the set of vertices, in which the vertex v_0 denotes the depot. E denotes a set of edges that connect the vertices. Each edge $e \in E$ has a cost $c(e) > 0$. If there is a task on an edge $e \in E$, then there is a positive demand $d(e)$, otherwise, $d(e) = 0$. The task set T is the ensemble of tasks, and the edges are positive demand. The objective of CARPs is to select the optimal set of routes for a number of vehicles, starting and ending at the depot v_0, to serve the set of tasks T while the total demand of tasks served on any route does not exceed a given vehicle capacity Q. The general formulation of CARP is explained in [10], [22].

B. D-MAENS

D-MAENS [16] uses a decomposition-based framework similar to MOEA/D [17], which divides the original CARP into many scalar subproblems with some uniformly distributed reference vectors. For a given objective vector $F(x) = (f_1(x), ..., f_n(x))$ and a reference vector $R = (r_1, ..., r_n)$, the expression for the objective function of a subproblem using weighted sum approach is as follows:

$$g^{ws}(x|R) = \sum_{i=1}^{n} r_i f_i(x). \tag{1}$$

For a bi-objective ($n = 2$) optimization problem, the reference vectors can be expressed as follows:

$$R^i = \left(\frac{i-1}{N-1}, \frac{N-i}{N-1} \right), \tag{2}$$

where N is the size of subproblems and $i \in [1, N]$. Every individual can be considered as a subproblem. Specifically, the current population is sorted in ascending order corresponding to the first objectives (total cost) of MO-CARP and the solutions with equal first objectives get sorted in descending order according to the second objectives (max total), which is also called representation and applied in line 12 of Algorithm 1. Moreover, each solution is maintained by a unique vector, which also means the number of subproblems is equal to that of reference vectors. In this way, we assign the current sorted population to subproblems $\{P_1, ..., P_N\}$. For given N reference vectors $\{R^1, ..., R^N\}$, a number of N SO-CARPs can be obtained by the original MO-CARP. From Equation 1, the objective function corresponding to the i-th subproblem is $g^{ws}(x|R^i)$.

D-MAENS is shown in Algorithm 1. First, population X and reference vectors $\{R^1, ..., R^N\}$ are initialized. Each solution $x \in X$ corresponds to a subproblem using the representation method introduced before. Then, neighboring subproblems are constructed based on the subproblems associated with corresponding reference vectors (line 6 in Algorithm 1). Second, for each subproblem $x_i^s \in X$, x_k^s and x_l^s are chosen from X_i, where X_i is the neighboring subproblems of subproblem x_i. Moreover, the crossover and local search of MAENS [5] are applied to these two individuals to generate solution y_i. $F(y_i)$ is inserted into O and dominated solutions of O are deleted (line 21 in Algorithm 1). After N offspring are generated (line 25 in Algorithm 1), D-MAENS employs the framework of NSGA-II to select N best solutions from the combination of X and Y, which is shown in Algorithm 2. In more detail, the fast nondominated sorting method is employed to evaluate $X \cup Y$. All the individuals in the first $k-1$ fronts are chosen for the next generation and crowding deletion is applied to choose solutions with good diversity in k-th front $Front_k$, where k is the minimum number s.t. $|\cup_{i=1}^{k} Front_i| \geq N$. Computation of the crowding distance method in one generation is applied to preserve diversity only once.

1) Strength and Drawbacks of D-MAENS: D-MAENS successfully combines the advantages of the MAENS method for SO-CARP and NSGA-II for MOEA, which leads to a better performance in real applications. On one hand, choosing maximum cost as the second objective and minimizing it is more in line with the needs of real problems, which keeps the cost of each route as even as possible. Furthermore, nondominated solutions obtained by D-MAENS can give decision-makers more opportunities to choose according to different realistic situations. On the other hand, improving diversity is significant to search processing and contributes to generating better solutions in convergence and diversity. In NSGA-II of D-MAENS, the operation of deleting some individuals based on the degree of crowding is able to improve diversity not only in objective space but also in decision variable space of MO-CARPs.

However, D-MAENS still has drawbacks which will be explained in three aspects. First, D-MAENS requires three parameters, the probability of crossover Pro_c and local search Pro_m, and the size of the neighboring subproblems T, which remains unchanged during the algorithm process. However, if some adjustments are applied to the parameters when the algorithm falls into local optimum, the algorithm will possibly jump out of the premature state. The second drawback of D-MAENS is that the crowding degree by measuring distance in the objective space may fail to estimate the similarities among the different routes of solutions. Finally, since the crowding degree is sensitive to the current population by using NSGA-II, the degree of remaining individuals will change after deleting one solution with the worst diversity. But in this generation, NSGA-II will continue to delete individuals based on the

Algorithm 1 General Framework of D-MAENS.

Require: the original MO-CARP P, subproblem size N, neighborhood size T
Ensure: final solution set O

/$*$ Step 1 Initialization $*$/
$O = \emptyset$
$\{R^1, ..., R^N\} \leftarrow$ Generate N reference vectors
$X = \{x_1, ..., x_N\} \leftarrow$Randomly initialize population
Associate P_i with R^i, for $i \in [1, N]$
Get the neighborhood $B(i) = \{i_1, ..., i_T\}$ for each P_i, where $r^{i_1}, ..., r^{i_T}$ are T closest reference vectors to r^i based on the Euclidean distance
/$*$ Step 2 Searching Process $*$/
while termination criteria not fulfilled **do**
  $Y = \emptyset$
  /$*$ Step 2.1 Generate Offspring $*$/
  for each subproblem $x_i^s \in X$ **do**
    Assign each subproblem to a unique representation $x_i^s \in X$
    Construct neighboring subproblems $X_i = \{x_{i_1}^s, ..., x_{i_T}^s\}$
    Randomly choose two x_k^s and x_l^s from X_i
    if $Random < Pro_c$ **then**
      $y_i \leftarrow$ Crossover (x_k^s, x_l^s)
    end if
    if $Random < Pro_m$ **then**
      $y_i \leftarrow$ Local Search (y_i)
    end if
    $O \leftarrow$ Update $(O \cup y_i)$
    $Y = Y \cup y_i$
  end for
  /$*$ Step 2.2 Selection of NSGAII $*$/
  $X \leftarrow$ perform Environmental Selection based on $(Y \cup X)$
end while

Algorithm 2 Environmental Selection of NSGA-II in D-MAENS.

Require: combined population P, size of population N
Ensure: output population O

1: $Front \leftarrow$ Perform nondominated sort based on P
2: $k \leftarrow$ minimum number satisfies $|\cup_{i=1}^{k} Front_i| \geq N$
3: $H \leftarrow |\cup_{i=1}^{k-1} Front_i|$
4: $B \leftarrow$ Perform crowding deletion based on $Front_k$
5: $O \leftarrow H \cup B$

wrong crowding degree, which may remove the individuals with good diversity and lead to inefficiency of maintaining the diversity.

III. D-MAENS2

In order to solve MO-CARPs with better performance, only making parameters constant and operating in the objective space is insufficient. We observe that when the number of non-dominated solutions in the combined population P is greater than N, which means $k = 1$ in line 2 of Algorithm 2, the performance of the algorithm is difficult to improve further. To address this problem, we adopt a strategy to adaptively adjust the parameters of D-MAENS. Moreover, we propose a novel method to measure similarities among solutions and adopt a proper way to remove the individuals based on the similarities. It's worth noting that the process of D-MAENS2 is the same as that of D-MAENS until the number of non-dominated solutions is greater than or equal to N. These two improvements will be explained in Sections III-A and III-B in detail.

A. Adaptive parameter modification

When the number of nondominated solutions is larger than or equal to N, the search process may trap in local optima and tend to increase the probability of local search Pro_m and the size of neighboring subproblems T through multiplying by different adjustment factors separately. There will be some overlapping solutions in objective space. Therefore we call those individuals having the same total cost and maximum cost a cluster. We denote the number of clusters as E and the number of individuals in the cluster who has the largest solutions as J. So in D-MAENS2, the probability of local search is changed into:

$$Pro_m = max\{Pro_m^{max}, Pro_m * \frac{N}{E}\}, \tag{3}$$

where Pro_m^{max} is the manually set upper limitation of the parameter. The size of neighboring subproblems is transformed into $max\{J, T\}$, where T is the parameter we predefined.

From the perspective of Pro_m, we consider that the smaller the number of clusters is, the worse the current population's diversity is. Therefore, we multiply the original parameters by $\frac{N}{E}$. Regarding to T, our goal is to enable individuals in the cluster with the largest size to crossover with individuals in other clusters.

B. Diversity Preservation

In D-MAENS, SO-CARP is transformed into MO-CARP by adding an additional objective, and the framework of NSGA-II is applied to maintain diversity. However, the crowding degree is evaluated only once and as the study [19] claimed that the crowding degree will change after deleting an individual if the degree strongly depends on the current population. Inspired by the archive truncation operation of SPEA2 [23], we adopt a strategy, iteratively deleting the solution with the worst diversity degree and updating the diversity degrees of all the remaining solutions. Moreover, since there is less information in bi-objective space, specifically, total cost and maximum cost, we propose a novel method to measure diversity degree in decision variable space instead of objective space whose idea is similar to the study of [24]. The detail is shown in Algorithm 3. If $k \neq 1$, the processing is the same as that of D-MAENS operation, otherwise, our designed method is adopted. First, build a gene pool $GPool$ based on the current population, where Ta is the number of tasks and formulation of $GPool$ is defined as:

$$GPool(i,j) = \sum_{n=1}^{N} \sum_{d=1}^{l_n-1} I(t_d^n == i \wedge t_{d+1}^n == j), \quad (4)$$

where i and j are different tasks; $I(a) = 1$ if a is true, otherwise, $I(a) = 0$; and the n-th solution in population is donated as $S_n = (t_1^n, ..., t_{l_n}^n)$, t_e^n is the e-th position of tasks or depot and l_n is the number of tasks including depot. Based on the $GPool$, the diversity of individual p corresponding to the current population is $Div(p, P)$, which is defined as:

$$Div(p,P) = \sum_{d=1}^{l_n-1} GPool(t_d^n, t_{d+1}^n). \quad (5)$$

Our purpose of measuring $GPool$ is to record the frequency of two tasks in pair completed in turn in all routes of all current solutions. For example, if the value of $GPool(i,j)$ is very high, we assume that there are numerous solutions which directly serve task j after completing task i. Then, on the basis of $GPool$, we count the total frequency of the sequence of each solution serving all the tasks. So the smaller the value Div is, the better diversity to be claimed is.

Algorithm 3 Environmental Selection in D-MAENS2.

Require: combined population P, size of population N
Ensure: output population O

```
Front ← Nondominated Sort (P)
k ← minimum number satisfies |∪_{i=1}^{k} Front_i| ≥ N
if k ≠ 1 then
    H ← |∪_{i=1}^{k-1} Front_i| ≥ N
    B ← Crowding Deletion (Front_k)
    O ← H ∪ B
else
    O ← Front_1
    while |O| ≠ N do
        GPool ← Construct Gene Pool(O)
        Div ← Calculate Diversity GPool
        p ← argmax_{p∈O} Div(p, P)
        O = O/{p}
    end while
end if
```

IV. Computational Studies

A. Experimental Setup

In order to evaluate the performance of D-MAENS2, we tend to compare D-MAENS2 with D-MAENS on three widely used benchmark sets, the gdb set [25], the val set [26] and the egl set [27]–[29]. The gdb test suite is proposed by DeArmon in [25] and has 23 instances with small-size. The val set was generated by Benavent [26] and contains 34 instances having 10 different graphs. Different instance of each graph is designed with different capacity of vehicles. The egl set was proposed by Eglese based on the application of gritting in Lancashire [27]–[29], which contains 24 instances and is based on two graphs. Among these three test suites, egl has the largest scale sizes of problems. To summarize, these three benchmark instances we applied have strong practicability, and the difficulty of the problem ranges from simple to hard. The key parameter settings are the same as the original D-MAENS, which is adopted in D-MAENS2 as initial parameters and is shown in Table I. All the results are run for 30 times independently on each test instance.

TABLE I
Parameter settings of the compared algorithms.

Parameter	D-MAENS2	D-MAENS
Population	60	60
Crossover rate	1	1
Mutation/LS rate	Pro_m^{max} =0.5	0.1
Max. generations	1000	1000
Neighborhood size	$T = 9$	9

B. Performance Measures

In this paper, we use three types of performance measures, namely, I_D, I_H, and I_E. The goal of designing indicators is similar to that in MOEAs, convergence and diversity. I_D and I_H are respectively modified variants of IGD [30] and HV [31] according to the particular multi-objective optimization problem, MO-CARPs, where IGD and HV are two well-known indicators to evaluate performances of both convergence and diversity in the field of MOEAs. In general, the high diversity of solutions in the objective space can roughly be considered as the better diversity in the decision space. Moreover, I_E considers more from SO-CARP. The details of indicators are as follows:

1) Distance From Reference Set (I_D): The indicator was proposed by [32] and formulated as:

$$I_D(A) = \frac{\sum_{y \in R} \min_{x \in A} d(x,y)}{|R|}. \quad (6)$$

$I_D(A)$ considers the average Euclidean distance from each solution in set R to its closet solution in A, where a smaller $I_D(A)$ means the distribution of A is more similar to R. Normally in MOEAs, R is set to the uniform sampling solutions of Pareto optimal solutions. Since obtaining R is very hard in MO-CARP, we construct R by combining all the nondominated solutions of all the compared algorithms over all runs of each instance in our experiments.

2) Hypervolume (I_H): This metric was suggested by [23] and its formulation is as follows:

$$I_H(A) = \int \cdots \int_{z \in \cup_{x \in A} HV(f(x), f^*)} 1 \cdot dz, \quad (7)$$

where $HV(f(x), f^*) = [f_1(x), f_1^*] \times \cdots \times [f_m(x), f_m^*]$ is the Cartesian product of the closed intervals $[f_i(x), f_i^*]$, $i = 1, ..., m$. I_H figures out the hypervolume (area in bi-objective space) which is dominated by at least one solution of the nondominated set and the details of computation are the same as D-MAENS. The larger the value I_H is, the better overall performance to be claimed is. It's worth noting that the two objectives should be normalized by A.

3) Elite (I_E): This metric is to compare the dominance relationship between the solutions, namely $elite$ solutions, with the least total cost of all the running results obtained by different algorithms in solving an instance. If the solution $elite_1$ of one algorithm weakly dominates the other solution $elite_2$, then this algorithm is considered to be better in dealing with this problem.

C. Experimental Results

Tables II, III and IV show the average values of I_D and I_H, respectively, over 30 independent runs of D-MAENS and D-MAENS2 and I_E on gdb, val and egl instances, where $|V|$, $|E|$ and $|R|$ mean the number of vertices, edges and tasks in the graph of the instance, respectively. τ stands for the least amount of vehicles required subject to the capacity constraint. For the gdb and val instances, it holds that $|E| = |R|$. The complexity of the instances increases as τ increases. The Wilcoxon rank sum-test with a significance level of 0.05 is carried on performing statistical analysis on the experimental results and the best values are highlighted in gray.

In gdb and val sets, the results of I_H and I_D are consistent that D-MAENS2 achieves statistically better values in 18 out of 23 in gdb test problems, 30 out of 34 in val test problems and other instances are all statistically similar. As to metric I_E, the two algorithms have the same results in most sets and D-MAENS2 has a slightly better solution in $gdb22$, $val1C$, $val4C$, $val8C$ and $val10D$ instances since D-MAENS already performed relatively well in gdb and val. Although the performance improvement on I_H is very slight, through I_H and I_D we can analyze that the diversity of the original D-MAENS is improved greatly based on our strategy.

In egl set, a large-scale test suite, the performance is improved significantly. In terms of I_D, D-MAENS2 outperforms D-MAENS in all instances. Moreover, regarding to I_H, D-MAENS2 achieves statistically better values in 33 out of 34. Observed on I_E, the results of egl are different from the results of gdb and val, and D-MAENS2 has been further improved on the base of D-MAENS.

For further observations, Fig. 1 plots HV and IGD average values of D-MAENS (red lines) and D-MAENS2 (blue lines) running 30 times on the $gdb13$, $val3C$, $E4-A$ and $S2-A$ problems, respectively, in 100, 200,...,1000 generations, where the nondominated solutions obtained by two algorithms in 1000 generation in each problem are considered as the reference set in the computation of I_D and I_H. Fig. 2 shows the nondominated solutions obtained by all 30 runs in $E4-A$, $S1-C$ and $S2-A$, respectively.

When solving $gdb13$ problem, although the two algorithms have the same performance in I_E (shown in Table II), we can see that the solution set obtained by D-MAENS2 is better than that obtained by D-MAENS in terms of I_H and I_D and the performance improvement of D-MAENS2 is more significant (in Fig. 1). In solving $val3C$ instance, although the final comparison results of I_H, I_D and I_E show that the two algorithms have the same ability in dealing with problems, D-MAENS2 converges faster. In dealing with $S1-C$ problem, the two algorithms have the same I_E values. However, the performance of I_H and I_D indicates that D-MAENS2 has better diversity and convergence from the perspective of MO-CARP. Fig. 2 also shows that all the nondominated solutions obtained by D-MAENS are dominated or weakly dominated by that obtained by D-MAENS2. In coping with $E4-A$ and $S2-A$ problems with large-size, the improvement of D-MAENS2 is also obvious. The performance of D-MAENS2 running in 200 generations is better than D-MAENs running 1000 generations observed by I_D and I_H in $E4-A$. Similarly, as shown in Fig. 2, compared with D-MAENS, D-MAENS2 generates more nondominated solutions which have better both convergence and diversity in $E4-A$.

V. Conclusion

In this paper, we propose D-MAENS2 and conduct a study on MO-CARPs to minimise the total cost and makespan. By analyzing the advantages and disadvantages of D-MAENS, we design two strategies, adaptive parameter adjustment by observing the state in objective space, and measuring and preserving diversity in decision variable space. Experimental studies on three famous CARP test suites indicate the superior performance of D-MAENS2 compared to the original D-MAENS.

When dealing with combinatorial optimization problems, decision makers tend to pay more attention to the diversity of solutions on the basis of acceptable costs. Therefore, we propose a novel way to measure and preserve diversity in the decision space. Experiments on the gdb, val and egl sets indicate that an improved D-MAENS with our strategy can also effectively find solutions with less cost in objective space. Moreover, our method is a general strategy, that can be applied to other algorithms.

References

[1] J. F. Campbell and A. Langevin, "Roadway snow and ice control," in *Arc Routing*. Springer, 2000, pp. 389–418.

[2] H. Handa, D. Lin, L. Chapman, and X. Yao, "Robust solution of salting route optimisation using evolutionary algorithms," in *2006 IEEE International Conference on Evolutionary Computation*. IEEE, 2006, pp. 3098–3105.

[3] F. Chu, N. Labadi, and C. Prins, "A scatter search for the periodic capacitated arc routing problem," *European Journal of Operational Research*, vol. 169, no. 2, pp. 586–605, 2006.

[4] B. L. Golden and R. T. Wong, "Capacitated arc routing problems," *Networks*, vol. 11, no. 3, pp. 305–315, 1981.

[5] K. Tang, Y. Mei, and X. Yao, "Memetic algorithm with extended neighborhood search for capacitated arc routing problems," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 1151–1166, 2009.

TABLE II

I_D, I_H AND I_E VALUES OF D-MAENS AND D-MAENS2 ANALYZED BY WILCOXON RANK SUM TEST ON *gdb* SET OVER 30 INDEPENDENT RUNS. THE BEST VALUES ARE HIGHLIGHTED IN GRAY. THE RESULTS INDICATE THAT OUR PROPOSED METHOD OUTPERFORMS D-MAENS2 ON MOST INSTANCES.

Instance	$\lvert V\rvert$	$\lvert E\rvert$	τ	I_D		I_H		D-MAENS2		D-MAENS	
				D-MAENS2	D-MAENS	D-MAENS2	D-MAENS	Total Cost	Makespan	Total Cost	Makespan
gdb1	12	22	5	0	0	0.029411	0.029411	316	74	316	74
gdb2	12	26	6	0.03923	0.111648	0.04553	0.04507	339	69	339	69
gdb3	12	22	5	0	0.007777	0.041322	0.041277	275	65	275	65
gdb4	11	19	4	0	0.000694	0.042733	0.04272	287	74	287	74
gdb5	13	26	6	0.00145	0.022239	0.04993	0.049738	377	78	377	78
gdb6	12	22	5	0	0.034535	0.04159	0.041266	298	75	298	75
gdb7	12	22	5	0	0.000556	0.051856	0.051831	325	68	325	68
gdb8	27	46	10	0.04326	0.114408	0.03517	0.033773	348	44	348	44
gdb9	27	51	10	0.03538	0.136089	0.0296	0.027751	303	43	303	43
gdb10	12	25	4	0.00262	0.012175	0.15888	0.158209	275	70	275	70
gdb11	22	45	5	0.01824	0.033329	0.16553	0.163863	395	80	395	80
gdb12	13	23	7	0.00676	0.0392	0.02639	0.026351	458	97	458	97
gdb13	10	28	6	0.45106	0.61739	0.02118	0.020913	536	151	536	151
gdb14	7	21	5	0	0.028717	0.09542	0.094389	100	21	100	21
gdb15	7	21	4	0.01075	0.028751	0.0873	0.084794	58	15	58	15
gdb16	8	28	5	0.01196	0.057187	0.10175	0.098691	127	26	127	26
gdb17	8	28	5	0.15052	0.250209	0.04246	0.036432	91	13	91	13
gdb18	9	36	5	0.01397	0.057559	0.09858	0.094654	164	33	164	33
gdb19	8	11	3	0	0	0.034501	0.034501	55	21	55	21
gdb20	11	22	4	0.01518	0.064014	0.06529	0.064718	121	34	121	34
gdb21	11	33	6	0.03412	0.073168	0.11005	0.106896	156	27	156	27
gdb22	11	44	8	0.09265	0.163997	0.0538	0.047895	200	25	200	26
gdb23	11	55	10	0.13422	0.198638	0.04575	0.039995	233	25	233	25

TABLE III

I_D, I_H AND I_E VALUES OF D-MAENS AND D-MAENS2 ANALYZED BY WILCOXON RANK SUM TEST ON *val* SET OVER 30 INDEPENDENT RUNS. THE BEST VALUES ARE HIGHLIGHTED IN GRAY. THE RESULTS INDICATE THAT OUR PROPOSED METHOD OUTPERFORMS D-MAENS2 ON MOST INSTANCES.

Instance	$\lvert V\rvert$	$\lvert E\rvert$	τ	I_D		I_H		D-MAENS2		D-MAENS	
				D-MAENS2	D-MAENS	D-MAENS2	D-MAENS	Total Cost	Makespan	Total Cost	Makespan
val1A	24	39	2	0.0055	0.021781	0.08501	0.084339	173	58	173	58
val1B	24	39	3	0.02355	0.044871	0.084	0.083028	173	58	173	59
val1C	24	39	8	0	0.026808	0.01128	0.011233	245	41	245	41
val2A	24	34	2	0.00897	0.019585	0.15704	0.156739	227	114	227	114
val2B	24	34	3	0.00561	0.016199	0.136	0.135811	259	108	259	108
val2C	24	34	8	1.47185	2.612839	0.008005	0.007795	457	71	457	71
val3A	24	35	2	0	0.002481	0.09415	0.094089	81	41	81	41
val3B	24	35	3	0	0	0.043216	0.043216	87	32	87	32
val3C	24	35	7	0	0	0.008264	0.008264	138	27	138	27
val4A	41	69	3	0.02282	0.044976	0.08671	0.085221	400	134	400	134
val4B	41	69	4	0.03369	0.068701	0.04781	0.046423	412	103	412	103
val4C	41	69	5	0.05812	0.102407	0.03888	0.037467	428	99	428	100
val4D	41	69	9	0.94954	1.600999	0.01001	0.009335	530	82	530	82
val5A	34	65	3	0.02611	0.039862	0.18587	0.184013	423	141	423	141
val5B	34	65	4	0.02908	0.050313	0.12598	0.123843	446	112	446	112
val5C	34	65	5	0.03466	0.062621	0.085	0.082619	474	95	474	95
val5D	34	65	9	0.14573	0.242464	0.02957	0.027888	581	80	581	80
val6A	31	50	3	0.00618	0.018546	0.11844	0.11776	223	75	223	75
val6B	31	50	4	0.01108	0.023717	0.09569	0.094985	233	68	233	68
val6C	31	50	10	0.03832	0.297183	0.02436	0.023176	317	54	317	54
val7A	40	66	3	0.00621	0.030095	0.17473	0.173432	279	85	279	85
val7B	40	66	4	0.00884	0.039536	0.10049	0.09891	283	58	283	58
val7C	40	66	9	0.02483	0.068382	0.05446	0.053697	334	50	334	50
val8A	30	63	3	0.0219	0.03565	0.16966	0.167445	386	129	386	129
val8B	30	63	4	0.03075	0.046578	0.11173	0.109005	395	99	395	99
val8C	30	63	9	0.13598	0.218798	0.03131	0.028797	522	78	525	79
val9A	50	92	3	0.03004	0.057283	0.16128	0.157332	323	108	323	108
val9B	50	92	4	0.03806	0.065405	0.1224	0.11754	326	82	326	82
val9C	50	92	5	0.05268	0.081589	0.08833	0.084263	332	67	332	67
val9D	50	92	10	0.0898	0.175132	0.02522	0.023833	391	49	391	49
val10A	50	97	3	0.04341	0.060297	0.25167	0.246307	428	143	428	143
val10B	50	97	4	0.04532	0.07567	0.21567	0.206615	436	109	436	109
val10C	50	97	5	0.03728	0.065942	0.14073	0.134647	446	90	446	90
val10D	50	97	10	0.08243	0.113862	0.0675	0.063233	529	62	530	62

TABLE IV

I_D, I_H AND I_E VALUES OF D-MAENS AND D-MAENS2 ANALYZED BY WILCOXON RANK SUM TEST ON *egl* SET OVER 30 INDEPENDENT RUNS. THE BEST VALUES ARE HIGHLIGHTED IN GRAY. THE RESULTS INDICATE THAT OUR PROPOSED METHOD OUTPERFORMS D-MAENS2 ON MOST INSTANCES.

Instance	$\|V\|$	$\|E\|$	$\|R\|$	τ	I_D		I_H		D-MAENS2		D-MAENS	
					D-MAENS2	D-MAENS	D-MAENS2	D-MAENS	Total Cost	Makespan	Total Cost	Makespan
E1-A	77	98	51	5	0.05685	0.099439	0.02975	0.029415	3548	943	3548	943
E1-B	77	98	51	7	0.26295	0.489896	0.01596	0.015684	4525	839	4516	899
E1-C	77	98	51	10	0.12821	0.8646	0.01065	0.009748	5595	836	5595	836
E2-A	77	98	72	7	0.00747	0.046029	0.04745	0.047038	5018	953	5018	953
E2-B	77	98	72	10	0.08225	0.134086	0.02084	0.020373	6317	878	6340	864
E2-C	77	98	72	14	0.14351	0.203078	0.0137	0.013426	8335	854	8335	854
E3-A	77	98	87	8	0.03664	0.067088	0.04296	0.042108	5898	929	5898	929
E3-B	77	98	87	12	0.08874	0.216701	0.01753	0.016602	7775	872	7779	872
E3-C	77	98	87	17	0.73591	1.609305	0.00902	0.008607	10303	827	10305	872
E4-A	77	98	98	9	0.05997	0.142883	0.03378	0.032618	6461	929	6470	929
E4-B	77	98	98	14	0.15812	0.215197	0.01926	0.018807	9007	914	9031	853
E4-C	77	98	98	19	3.73255	10.63184	0.01269	0.011649	11614	872	11632	872
S1-A	140	190	75	7	0.02209	0.030614	0.06348	0.063392	5018	1023	5018	1023
S1-B	140	190	75	10	0.03861	0.072428	0.03395	0.033423	6394	984	6394	984
S1-C	140	190	75	14	0.04029	0.131	0.025	0.024579	8518	1018	8518	1018
S2-A	140	190	147	14	0.07788	0.110297	0.03057	0.029565	9920	1072	10034	1076
S2-B	140	190	147	20	0.14702	0.244631	0.0178	0.016712	13223	1040	13294	1040
S2-C	140	190	147	27	0.34319	0.855082	0.01374	0.012675	16510	1040	16596	1040
S3-A	140	190	159	15	0.0605	0.087608	0.03326	0.032302	10325	1064	10333	1065
S3-B	140	190	159	22	0.13116	0.215597	0.01907	0.018381	13818	1040	13837	1106
S3-C	140	190	159	29	0.40335	1.121586	0.01335	0.0122	17255	1040	17357	1040
S4-A	140	190	190	19	0.42372	0.746238	0.01117	0.010458	12341	1060	12425	1060
S4-B	140	190	190	27	93.2086	204.4921	0.00779	0.007226	16358	1027	16468	1040
S4-C	140	190	190	35	234.339	368.4267	0.00732	0.00678	20579	1027	20829	1027

[6] Y. Mei, X. Li, and X. Yao, "Decomposing large-scale capacitated arc routing problems using a random route grouping method," in *Evolutionary Computation (CEC), 2013 IEEE Congress on*. IEEE, 2013, pp. 1013–1020.

[7] ——, "Cooperative coevolution with route distance grouping for large-scale capacitated arc routing problems," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 3, pp. 435–449, 2014.

[8] ——, "Variable neighborhood decomposition for large scale capacitated arc routing problem," in *Evolutionary Computation (CEC), 2014 IEEE Congress on*. IEEE, 2014, pp. 1313–1320.

[9] J. Liu and X. Yao, "Self-adaptive decomposition and incremental hyper-parameter tuning across multiple problems," in *2019 IEEE Symposium Series on Computational Intelligence*, 2019, pp. 1590–1597.

[10] K. Tang, J. Wang, X. Li, and X. Yao, "A scalable approach to capacitated arc routing problems based on hierarchical decomposition," *IEEE Transactions on Cybernetics*, vol. 47, no. 11, pp. 3928–3940, 2017.

[11] A. Hertz, G. Laporte, and M. Mittaz, "A tabu search heuristic for the capacitated arc routing problem," *Operations Research*, vol. 48, no. 1, pp. 129–135, 2000.

[12] P. Beullens, L. Muyldermans, D. Cattrysse, and D. Van Oudheusden, "A guided local search heuristic for the capacitated arc routing problem," *European Journal of Operational Research*, vol. 147, no. 3, pp. 629–643, 2003.

[13] P. Greistorfer, "A tabu scatter search metaheuristic for the arc routing problem," *Computers & Industrial Engineering*, vol. 44, no. 2, pp. 249–266, 2003.

[14] P. Lacomme, C. Prins, and W. Ramdane-Cherif, "Competitive memetic algorithms for arc routing problems," *Annals of Operations Research*, vol. 131, no. 1-4, pp. 159–185, 2004.

[15] P. Lacomme, C. Prins, and M. Sevaux, "A genetic algorithm for a bi-objective capacitated arc routing problem," *Computers & Operations Research*, vol. 33, no. 12, pp. 3473–3493, 2006.

[16] Y. Mei, K. Tang, and X. Yao, "Decomposition-based memetic algorithm for multiobjective capacitated arc routing problem," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 2, pp. 151–165, 2011.

[17] Q. Zhang and H. Li, "Moea/d: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on evolutionary computation*, vol. 11, no. 6, pp. 712–731, 2007.

[18] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[19] Y. Tian, C. He, R. Cheng, and X. Zhang, "A multistage evolutionary algorithm for better diversity preservation in multiobjective optimization," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–15, 2019.

[20] M. Kim, T. Hiroyasu, M. Miki, and S. Watanabe, "Spea2+: Improving the performance of the strength pareto evolutionary algorithm 2," in *Parallel Problem Solving from Nature - PPSN VIII*, X. Yao, E. K. Burke, J. A. Lozano, J. Smith, J. J. Merelo-Guervós, J. A. Bullinaria, J. E. Rowe, P. Tiňo, A. Kabán, and H.-P. Schwefel, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 742–751.

[21] G. B. Dantzig and J. H. Ramser, "The truck dispatching problem," *Management Science*, vol. 6, no. 1, pp. 80–91, 1959.

[22] J. Liu, K. Tang, and X. Yao, "Robust optimisation in uncertain capacitated arc routing problems: Progresses and perspectives," 2020.

[23] E. Zitzler, M. Laumanns, and L. Thiele, "Spea2: Improving the strength pareto evolutionary algorithm," *TIK-report*, vol. 103, 2001.

[24] O. Cuate and O. Schütze, "Variation rate to maintain diversity in decision space within multi-objective evolutionary algorithms," *Mathematical and Computational Applications*, vol. 24, no. 3, p. 82, 2019.

[25] J. S. DeArmon, "A comparison of heuristics for the capacitated chinese postman problem," Ph.D. dissertation, University of Maryland, 1981.

[26] E. Benavent, V. Campos, A. Corberán, and E. Mota, "The capacitated arc routing problem: lower bounds," *Networks*, vol. 22, no. 7, pp. 669–690, 1992.

[27] R. W. Eglese, "Routeing winter gritting vehicles," *Discrete applied mathematics*, vol. 48, no. 3, pp. 231–244, 1994.

[28] R. W. Eglese and L. Y. Li, "A tabu search based heuristic for arc routing with a capacity constraint and time deadline," in *Meta-Heuristics*. Springer, 1996, pp. 633–649.

[29] L. Y. Li and R. W. Eglese, "An interactive algorithm for vehicle routeing for winter—gritting," *Journal of the Operational Research Society*, vol. 47, no. 2, pp. 217–228, 1996.

[30] A. Zhou, Y. Jin, Q. Zhang, B. Sendhoff, and E. Tsang, "Combining model-based and genetics-based offspring generation for multi-objective optimization using a convergence criterion," in *2006 IEEE International Conference on Evolutionary Computation*. IEEE, 2006, pp. 892–899.

[31] L. While, P. Hingston, L. Barone, and S. Huband, "A faster algorithm for calculating hypervolume," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 1, pp. 29–38, 2006.

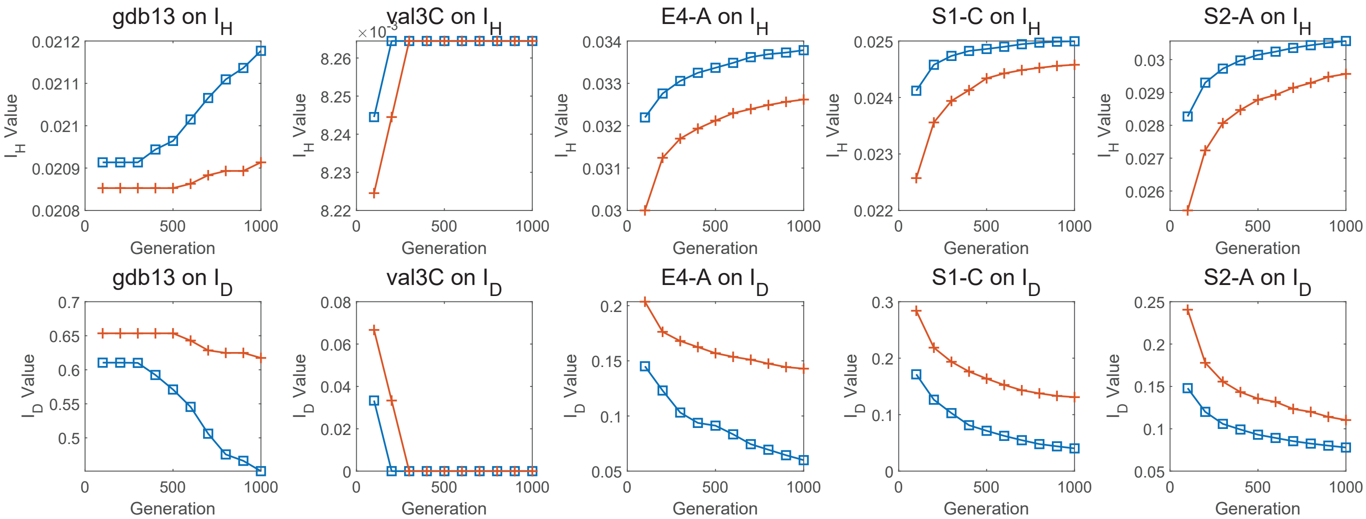

Fig. 1. Curves of HV and IGD average values of D-MAENS (red lines) and D-MAENS2 (blue lines) running 30 times on the $gdb13$, $val3C$, $E4-A$ and $S2-A$ problems, respectively, in 100, 200,...,1000 generations.

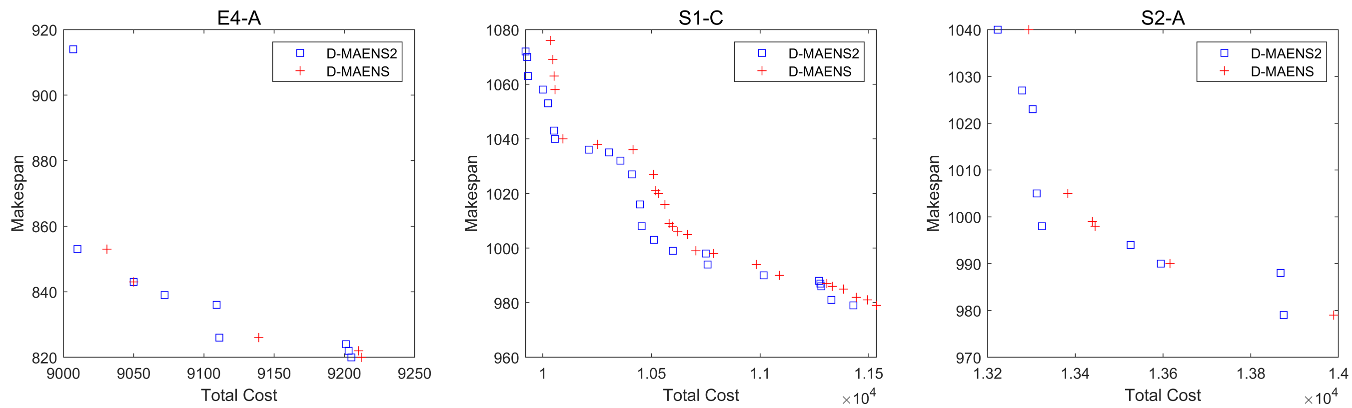

Fig. 2. Nondominated solutions generated by all 30 runs of D-MAENS and D-MAENS2 on $E4-A$, $S1-C$ and $S2-A$ of egl set.

[32] P. Czyżak and A. Jaszkiewicz, “Pareto simulated annealing—a meta-heuristic technique for multiple-objective combinatorial optimization,” *Journal of Multi-Criteria Decision Analysis*, vol. 7, no. 1, pp. 34–47, 1998.