Soft Comput (2011) 15:149–182
DOI 10.1007/s00500-010-0543-9

ORIGINAL PAPER

ϵ-DANTE: an ant colony oriented depth search procedure

Pedro Cardoso · Mário Jesus · Alberto Márquez

Published online: 3 February 2010

Abstract The ϵ-Depth ANT Explorer (ϵ-DANTE) algorithm applied to a multiple objective optimization problem is presented in this paper. This method is a hybridization of the ant colony optimization algorithm with a depth search procedure, putting together an oriented/limited depth search. A particular design of the pheromone set of rules is suggested for these kinds of optimization problems, which are an adaptation of the single objective case. Six versions with incremental features are presented as an evolutive path, beginning in a single colony approach, where no depth search is applied, to the final ϵ-DANTE. Versions are compared among themselves in a set of instances of the multiple objective Traveling Salesman Problem. Finally, our best version of ϵ-DANTE is compared with several established heuristics in the field showing some promising results.

Keywords Swarm intelligence optimization · Ant colony optimization · Hybrid algorithms · Multiple objective optimization · Depth local search

P. Cardoso (✉) · M. Jesus
University of Algarve, ISE, Campus da Penha,
8005-139 Faro, Portugal
e-mail: pcardoso@ualg.pt

M. Jesus
e-mail: mjesus@ualg.pt

A. Márquez
University of Sevilla,
ESTII Avda. Reina Mercedes,
41012 Sevilla, Spain
e-mail: almar@us.es

1 Introduction

Nowadays, decision processes relies, more than ever, on results produced by special skilled techniques to approximate solutions for complex problems. Among these there is a class where a trade-off between several objectives is required. They are known as multi-objective optimization (MOO) problems. To understand the intrinsic behavior and relations between objectives, the assistance of the Pareto principle is required (Deb 2001).

In this article, an ant colony optimization algorithm (ACO) for the multi-objective optimization problems is adapted in a suitable way to solve the multiple objective Traveling Salesman Problem (MOTSP). It includes a limited depth search in order to improve the performance of the method and is tested on several instances of the MOTSP. This combinatorial problem, as well as the majority of the MOO problems (see for example the Multiple Objective Minimum Spanning Trees problem case in (Camerini et al. 1980, 1983; Hamacher and Ruhe 1994) are classified as NP-complete and NP-# (Ehrgott and Gandibleux 2004; Garey and Johnson 1990).

Some examples of methods developed to solve MOO problems focus on the transformation of the problem from the multi into a single objective problem, aiming at the use of the optimization techniques developed for the later case. Some examples are the Weighted Sum, the ϵ-Constraint, and the Weighted Metric methods (Miettinen 1999). However, those methods have some important weaknesses, as the eventuality of the subsequent single objective optimization problem has not an efficient algorithm to solve it (Knowles and Corne 2001) (which is exactly the case of the MOTSP), or the impossibility to return efficient solutions in a concave region of the Pareto front and, therefore, the incapability of assuring a well-distributed/representative

set of solutions (Deb 2001; Hamacher and Ruhe 1994). Other problems can occur like the possible non-optimality of the solutions returned by the ϵ-Constraint method (Miettinen 1999).

Tailored with particular insight in nature phenomena there is a set of methods which are recognized as approximation methods or meta-heuristics. Most of these methods were first introduced to solve single objective optimization problems (Aarts and Lenstra 1997), but, due to their characteristics, they were more or less straightforwardly adapted to the multiple objective case. These methods abstractly describe an optimization concept characterized by a set of common optimization steps to be performed independently of the problem, making them generally applicable and flexible. In general, those meta-heuristics only require a set of feasible solutions, a weight function, a neighborhood operator, and an efficient method to explore that neighborhood. Some of these meta-heuristics are the well-known Tabu Search (TS) (Glover and Laguna 1997), Simulated Annealing (SA) (Kirkpatrick et al. 1983), Genetic Algorithms (GA) (Parmee 2001), and the Swarm Intelligence (SI) algorithms (Dorigo and Stützle 2004).

We are mainly focused on ACO algorithms which is an important subclass of the SI algorithms. ACO algorithms are based on the behavior of the ants colonies, and, as in other meta-heuristics, it is possible to use different characteristics according to the problem to be solved. For instance, we can consider single/multiple colonies, single/multiple pheromone matrices, elitist and non-elitist pheromone updating rules, and updating rules that allow (or not) ants to update pheromone trails of other colonies (García-Martínez et al. 2004, 2007). The SI class of algorithms are known to produce good approximations, within limited computational requirements. However, they usually need a large amount of computational time or resources to be competitive, specially when an earlier obtained solution needs to be improved (Dorigo et al. 1999).

Those circumstances make the use of hybrid algorithms a possible solution to refine the results obtained with SI methods (Dorigo and Stützle 2004; Paquete and Stützle 2006). For instance, Reimann and Laumanns (2004) reported the study of an hybrid ACO algorithm for the Capacitated Minimum Spanning Tree problem. In this case, an ACO technique is applied in a first phase, which is followed by Prim's algorithm applied to clusters of nodes that were formerly computed. Blum (2005b) developed the Beam–ACO, which is a combination of a beam search heuristic with an ACO algorithm. In the Beam–ACO, the basic solution construction mechanism of standard ACO algorithms is replaced by a new one where each artificial ant performs a probabilistic beam search. Reimann et al. (2004) presented a Savings based Ant System for the Vehicle Routing Problem, where "ants" construct a solution which is improved by a local search. This local search used swap movements followed by a 2-opt algorithm.

This paper presents the DANTE and ϵ-DANTE algorithms. These algorithms appear as an effort to provide a powerful way of further exploiting the solutions built by an ACO algorithm. In DANTE all solutions enter in a limited depth search phase which is oriented by the pheromone trails. On the other hand, in ϵ-DANTE, only solutions that improve the approximation set or that are considered "attractive" enter the same depth search phase. More precisely, whenever a solution is inserted into the approximation set or satisfies an ϵ distance to the approximation set criterion, an oriented limited depth search is performed using the pheromone values to guide that search. These actions are an attempt to improve the exploitation of the search space, by conducting depth search to the more promising areas which, at the same time, positively affects its exploration.

Furthermore, the proposed pheromone updating strategy tries to avoid some noisy trails, that are present when all computed solutions are used in the updating rules. This is achieved by restricting the solutions that will contribute to the pheromone trails in each update to some subset of the approximation set. This strategy comes in the direction of the BS (best-so-far solution) update rule (Blum 2005a).

As already mentioned, the MOTSP was used to test our implementation of the ϵ-DANTE method. For the MOTSP problem there is a set of studies which is based on the mentioned meta-heuristics, as in the ACO algorithm (García-Martínez et al. 2004), in the GA (Jaszkiewicz 2002; Murata 1997), in the SA algorithm (Ulungu et al. 1999), or in the Pareto Local Searches (Paquete and Stützle 2003), which per se produces a base line for our methods.

We must notice that this paper is presented in a build-up way, as we depart from a basic version and, through consecutive improvements, end up with the final ϵ-DANTE. More precisely, six versions are studied. The first three versions are general ACO methods: starting from a single colony method (Version 0), which passes to a multiple equal behavior colony version (Version 1), evolving to a multiple with distinct behavior colony version (Version 2). The remaining versions are hybridizations of the ACO algorithm with a depth search procedure, that come out as attempts to properly exploit the search space of the problems in study. Therefore, Versions 3 and 4 (DANTE and DANTE II) are first presented and significantly improve the previous results. ϵ-DANTE finishes our proposal of a limited oriented depth search method.

As an outline of this paper, some preliminaries and common features, among all versions, are given in the next section. The developments of the procedures are detailed in Sects. 3 and 4. A set of experimental results, including MOTSP instances with more than two objectives, are

resumed in the fifth section and in the next one some comparisons with other methods in the field are presented and discussed. This contribution ends with some considerations and some possible evolution for this investigation.

2 Preliminaries and algorithms common features

This section is devoted to present the fundamental definitions for the remaining paper. The section starts with the basic formulation of the MOO problems, and a brief overview of the ACO meta-heuristic applied to these problems. The common features of the proposed methods are presented in the remaining section.

2.1 Multiple objective optimization problems

The MOO problems are characterized by the existence of several objectives to be optimized. Commonly, these objectives are incompatible in the sense that to improve one of them, it is necessary to worsen at least one of the others (Deb 2001; Miettinen 1999; Romero 1993). Without lost of generality, MOO problems can be formalized as

$$\min_{X \in \mathcal{S}} \mathcal{W}(X), \tag{1}$$

where $\mathcal{W}(X) = (w_1(X), w_2(X), \ldots, w_n(X))$ is a vector objective function where all components, w_i, are to be minimized over the feasible set, $\mathcal{S}$.

The Pareto (or efficiency) order relation will be used to compare solutions. In this case, a solution X is said to dominate another solution Y, $X \prec Y$, when X is not worse than Y for all objectives and there is at least one on which it is strictly better, that is,

$$\begin{cases} \forall_{i \in \{1,2,\ldots,n\}} : w_i(X) \leq w_i(Y) \\ \exists_{j \in \{1,2,\ldots,n\}} : w_j(X) < w_j(Y). \end{cases} \tag{2}$$

A solution X weakly dominates a solution Y, $X \preceq Y$, if X is not worse than Y for all objectives, that is, $\forall_{i \in \{1,2,\ldots,n\}} : w_i(X) \leq w_i(Y)$. The solution of the MOO problem, called Pareto (or efficient) set, is the set of the non dominated solutions (also called Pareto solutions, optimal solutions, or efficient solutions).

Pragmatically, a meta-heuristic returns an approximation to the Pareto set, in the sense that it does not necessarily compute all solutions or all solutions are Pareto solutions. With the objective of comparing the performance of MOO meta-heuristics, several metrics were developed (Zitzler et al. 2000, 2003). In this paper we will use the C (set coverage), $\mathcal{S}$ (hyper-volume ratio), $R1$, and $R3$ metrics (see Appendix A for a brief description of these metrics).

In the remainder of the article, we will also consider that all problems can be formalized as networks in the form $\mathcal{N} = (\mathcal{V}, \mathcal{E}, \mathcal{Z})$, where $\mathcal{V}$ is the set of nodes, $\mathcal{E}$ is the set of edges, and $\mathcal{Z} : \mathcal{E} \rightarrow (\mathbb{R}^+)^m$ such that

$$\mathcal{Z}(e) = (z_1(e), z_2(e), \ldots, z_m(e))$$

is a function that associates to each edge a m-weight vector.

2.2 Multiple Objective Ant Colony Algorithms

Ant colony optimization algorithms are meta-heuristics based on the collective behavior of the majority of the ant colonies (Dorigo and Stützle 2004). In those colonies, ants use pheromones to communicate between them, which includes information about paths to known resources, perilous situations, or some other kind of information (Johnson 2001).

ACO algorithms mimic those behaviors using a set of agents that compute new solutions based on artificial pheromone trails left by the previous agents. Technically, these pheromone trails are numerical values reflecting the best solutions found so far.

ACO algorithms have a background of success solving many multiple objective optimization problems (García-Martínez et al. 2004).

The general process can be described as follows. For each cycle, a set of solutions based on the pheromone matrices are computed. These solutions are then evaluated and used to update the approximation set and the pheromone matrices. The overall procedure is supported by the positive and negative feedbacks generated by pheromone-updating strategies, which is the base of most of the ACO algorithms. Algorithm 1 sketches a general ACO.

Algorithm 1 General ACO Algorithm

```
1: Initialize the pheromone matrices and the approximation set, P = ∅
2: repeat
3:     for all ants do
4:         Compute a solution based on the pheromone trails and possible heuristics
5:         Evaluate the solution and update the approximation set
6:     end for
7:     Update the pheromone trails
8: until stopping criteria is met
9: return Approximation set.
```

Despite the general good performance of the ACO algorithms, along with the majority of the meta-heuristics, these methods have certain difficulties to achieve optimum solutions, although they return near optima (Dorigo et al. 1999). As an attempt to improve the methods efficiency, hybridizations is nowadays considered to be a fundamental aspect of high performing algorithms, as pointed out by Blum (2005a). Blum and Roli (2003) distinguish three

different taxonomies for hybridization: (1) the case where components from one meta-heuristic are included into another one; (2) the case where a cooperative search between systems is performed, based on an information exchange; and (3) the case where there is an integration of approximate and systematic (or complete) methods. An in-depth analysis is reported by Talbi (2002) where a complementary taxonomy for the hybrid meta-heuristics is presented.

Therefore, solutions with good objective values should not be simply evaluated and discarded. For several evolutionary algorithmic implementations, it is common to use hybrid algorithms that apply local optimizers to the solutions obtained by the constructive algorithms. This algorithmic technique allows the further exploration of the achieved solutions, as for example the 2-opt or 3-opt for the Traveling Salesman Problem (Dorigo and Stützle 2004; Paquete and Stützle 2006), the SOP-3-exchange for the Sequential Ordering Problem (Gambardella and Dorigo 2000), or the Iterated Local Search for the Bin Packing Problem (Levine and Ducatelle 2004).

A different approach for the traditional methods of local search, in order to improve the computational performance, could be centered in the exploration of the neighborhood of the more promising solutions, avoiding exploratory computations that probably would not improve the approximation set. This exploration should also be supported by the known information related to the problem and not just mechanical and non oriented exploration of the neighbors. One alternative found in the literature is the Beam-ACO (Blum et al. 2006; Blum 2005b) which, as already referred, merges a beam search with an ant colony strategy, obtaining an oriented/stochastic beam search.

2.3 Common features to all versions

Our proposal of a new stochastic depth search method will be presented in the next sections. The proposed methods are based on the available information, namely pheromone trails, to guide the limited depth search. The objective is to lead the exploration and the exploitation of the search space using the maintained information. Six versions will be presented, justifying as much as possible the evolution from one version into the other. The experimental methodology and the remaining common features are explained in the next paragraphs.

2.3.1 Updating the approximation set

The update of the approximation set is made in the classical way. Given a solution S and the approximation set $\mathcal{P}$, the update procedure inserts S into $\mathcal{P}$ iff S is not dominated by any solution of $\mathcal{P}$. Solutions in $\mathcal{P}$ which are dominated by S are removed. Furthermore, the merge of two approximation sets, $\mathcal{P}_1 \uplus \mathcal{P}_2$, is the result of keeping the non-dominated elements of $\mathcal{P}_1 \cup \mathcal{P}_2$.

2.3.2 Pheromone trails update

When referring to the pheromone matrices, two main approaches were previously considered: the use of a single pheromone matrix, and the use of m matrices each one related to one of the weights (García-Martínez et al. 2007).

In this article m matrices were considered, which is equivalent to assigning a m-pheromone vector to each edge e. The colony's pheromone-vector update is made according to formula

$$\tau(e) = \rho\tau(e) + \Delta(e), \quad e \in \mathcal{E}, \tag{3}$$

where $\tau(e) = (\tau_1(e), \tau_2(e), \ldots, \tau_m(e))$ is the m-pheromone vector associated to edge e;$\rho \in [0, 1]$ is called the persistence factor ($1 - \rho$ is the evaporation factor). The smaller the value of ρ, the smaller the quantity of information, used in one cycle, transmitted to the following cycle; $\Delta(e) = (\Delta_1(e), \Delta_2(e), \ldots, \Delta_m(e))$ is the reinforcement pheromone vector associated to edge e and is computed using the elements of the approximation set, $\mathcal{P}$, along with formula

$$\Delta_k(e) = \sum_{T \in \mathcal{P}} \frac{Q}{w_k(T)}, \quad k = 1, 2, \ldots, m \tag{4}$$

where Q is a value with the same magnitude of the solutions. For example, if the weights are balanced, Q can be set as the minimum value between all objectives; $\mathcal{P}$ is the colony's approximation set (further details about this set are given in the next section).

2.3.3 Transition rule

For all versions the order in which the nodes are chosen to belong to the Hamiltonian cycle is made pseudo randomly. This selection is made according to the pheromone trails and a local greedy heuristic that gives preference to the "nearest" nodes.

The choice of the edges to be included in the Hamiltonian cycle is made as follows. If the agent is on node s and T_N is the set of nodes already included in the path that is being constructed, then an edge e_{st} is chosen to integrate the path (and consequently node t is added) according to

$$e_{st} = \begin{cases} \arg\max_{e_{st'} \in \mathcal{A}_s} \prod_{k=1}^{m} \tau_k(e_{st'})^{\alpha_k} \eta_k(e_{st'})^{\beta_k} & \text{if } q \leq q_0 \\ e & \text{if } q > q_0, \end{cases} \tag{5}$$

where s is the previously inserted node; t is the node selected to be inserted in the path; $\mathcal{A}_s = \{e_{st'} \in \mathcal{E} : t' \notin T_N\}$ is the set of feasible edges with nodes adjacent to s;

$\tau_k(e)$ is the pheromone value associated to the k weight in edge e; $\eta_k(e) = \frac{1}{z_k(e)}$ implements the local greedy heuristic, where $z_k(e)$ is the k-weight of edge e; α_k is an algorithm parameter associated with the relevance of k-pheromone trail. Indirectly, this value is associated with the relevance of the k-weight since each pheromone trail is associated to a weight; β_k is an algorithm parameter associated to the local heuristic that favors edges with lower k-weight; $e \in \mathcal{A}_s$ is an edge pseudo randomly chosen with probability

$$p(e) = \frac{\prod_{k=1}^{m} \tau_k(e)^{\alpha_k} \eta_k(e)^{\beta_k}}{\sum_{f \in \mathcal{A}_s} \prod_{k=1}^{m} \tau_k(f)^{\alpha_k} \eta_k(f)^{\beta_k}}; \quad (6)$$

q is a uniform random value in [0, 1]; and $q_0 \in [0, 1]$ is a parameter that favors the exploration of the search space (for smaller values of q_0) or the exploitation of that same search space (for larger values of q_0).

2.4 Test problem: multiple objective Traveling Salesman Problem

A set of instances of the MOTSP were used to test the proposed methods. The MOTSP Problem requires the computation of the "cheapest" Hamiltonian cycles according to the Pareto principles. This choice is justified by the availability of well known sets of problems and respective solutions or approximations. In particular we have used the bi-objective kroAB50 problem[1] to test the upgrade features introduced from version to version. To compare the best performing variants a broader set of instances will be presented in a subsequent section.

2.5 Computational environment and performance metrics

The methods compared in this contribution were implemented in C++, compiled using gcc 4.3.3, and run on Ubuntu 7.10 over a 3.0 GHz computer with 1 GB of RAM.

To compare and test the performance of the proposed methods, five metrics were selected: ($|\cdot|$) cardinality of the approximation set, C (coverage), $\mathcal{S}$ (hyper-volume ratio), $R1_R$, and $R3_R$ (see Appendix A). The C metric is used to compare two sets, calculating the percentage of elements of one set which are weakly dominated by elements of the other, or vice versa. The hyper-volume ratio calculates the percentage of hyper-volume obtained with a certain method, when compared with the hyper-volume of a reference set. This metric allows us to measure the diversity of the approximation and its proximity to the reference set. The other two metrics, $R1_R$ and $R3_R$, use utility functions to measure the quality of the solutions when compared with a reference set. We recall that using $\mathcal{P}_0$ as the reference set, the objective is to obtain $S(\leq 1)$ values as close as possible to 1, $R1_r(\geq 0.5)$ as close as possible to 0.5 and $R3_R(\leq 0)$ as close as possible to 0.

Furthermore, we have used the non-dominated elements of the union of all runs and versions variants described in the document (after 1,800 s of computation time) as a reference set, $\mathcal{P}_0$. We should also notice that $\mathcal{P}_0$ has 847 elements and its hyper-volume is equal to 4.876.026.455 for the (anti-)reference solution (90.000, 90.000).[2]

Table 1 collects the remaining common parameters.[3] Parameters which are not common are to be presented with the correspondent versions. Since it is impossible to place in this paper all combinations of the parameters over large sets of values, the strategy was to establish constant suitable values for some of them, while varying others.

Table 1 Global parameter values for all test variants

	Parameter	Obs.
Ants per cycle	10	
Cycles per colony	50	See next sections
ρ	0.1	Eq. 3
Q	15,000	Eq. 4
$q0$	0.75	Eq. 5
Maximum time	300 s	Time used for the comparisons (unless stated otherwise)
Number of runs	25	

3 An initial approach to multiple objective ant colony optimization algorithms

The next sections present the six proposed methods. It starts with a base multi-objective approach and builds-up new methods, through successive improvements, until the final proposal.

[1] The used problems are combinations of the single objective problems with the same prefix (kro) available at the TSPLib. Files are truncated to obtain the 50 node instances, and pairwise combined to obtain the multiple objectives (Jaszkiewicz 2002).

[2] We notice that comparing $\mathcal{P}_0$ with non-dominated subset of the union of the results obtained with five runs for each of the Multiple Objective Genetic Local Search (MOGLS), Multiple Objective Simulated Annealing (MOSA), and MOSA-like MOGLS algorithms (Jaszkiewicz 2002, 2006; Ishibuchi and Murata 1998) [results available at (Jaszkiewicz 2006)],$\mathcal{P}'_0$, the C metric value were equal to $C(\mathcal{P}'_0, \mathcal{P}_0) = 0.34829$ and $C(\mathcal{P}_0, \mathcal{P}'_0) = 0.99517$.

[3] The total run time was 1,800 s although for the comparisons, and unless stated otherwise, we consider the 300 s computation time. The objective of using the 1,800 s running time was to obtain a more accurate approximation set to serve as a reference set.

3.1 Version 0: base version

For the base version, Version 0, the general single colony MOACO that was presented in Sect. 2 is considered as follows. Furthermore, a classical constructive process was used to build each TSP cycle. This construction can be described as follows:

1. Initially each agent is placed at random in one of the network nodes;
2. Then, while there are unvisited nodes, each of those agents builds a path by the successive addition of nodes from the ones that were not yet visited, and are adjacent to the one that was last inserted. The selection of the nodes is made pseudo-randomly using Eq. 5;
3. The cycle is closed by the addition of the edge defined by the first and last nodes inserted in the path.

The weight of a cycle is obtained by the sum of all edges weights, that is:

$$\mathcal{W}(P) = \sum_{e \in P} \mathcal{Z}(e) = \sum_{e \in P} (z_1(e), z_2(e), \ldots, z_m(e)) \quad (7)$$

3.1.1 A first set of results

With the parameter values summarized in Table 2, four variants were considered to test Version 0. Table 3 presents the statistical results for the number of non-dominated solutions (cardinality of the approximation set), $\mathcal{S}, R1_R$, and $R3_R$ metrics. Through the observation of the metric values, we can conclude that this version variants have a low performance. When compared with the reference set values, the highest value for the $\mathcal{S}$ metric is obtained for variant $A.1$ with 0.8601, corresponding to 86.01% of the hyper-volume of $\mathcal{P}_0$. Analogous conclusions can be derived from the $R1_R$ and $R3_R$ values, with values far from the objective 0.5 and 0, respectively.

Figure 1 sketches typical objective fronts for each of the variants and for the reference set, $\mathcal{P}_0$. It is observable that the variants fronts, compared with $\mathcal{P}_0$, do not achieve the proximity and diversity required in an multi-objective approximation. In the next section we will present a first modification with the objective of improving both diversity and proximity objectives.

Table 2 Parameter values for the tested variants of Versions 0

Variant	Colonies	α	β
A.1	1	1.0	1.0
A.2	1	5.0	1.0
A.3	1	5.0	5.0
A.4	1	1.0	5.0

A all variants of this version

Table 3 Version 0 statistics (mean and standard deviation) for the metrics: $|P|$, $\mathcal{S}, R1_R$, and $R3_R$

	A.1	A.2	A.3	A.4
$\|P\|$				
Mean	**1.427e+02**	4.252e+01	6.872e+01	1.266e+02
SD	2.510e+01	1.139e+01	2.351e+01	4.371e+01
$\mathcal{S}$				
Mean	**8.601e−01**	7.505e−01	7.679e−01	8.173e−01
SD	1.434e−02	1.468e−02	4.760e−02	7.676e−02
$R1_R$				
Mean	**9.984e−01**	1.000e+00	1.000e+00	9.992e−01
SD	6.308e−03	0.000e+00	0.000e+00	2.797e−03
$R3_R$				
Mean	**−5.141e+03**	−9.957e+03	−8.782e+03	−5.805e+03
SD	1.565e+03	1.132e+03	2.416e+03	4.271e+03

Best mean value for each case is shown in bold type

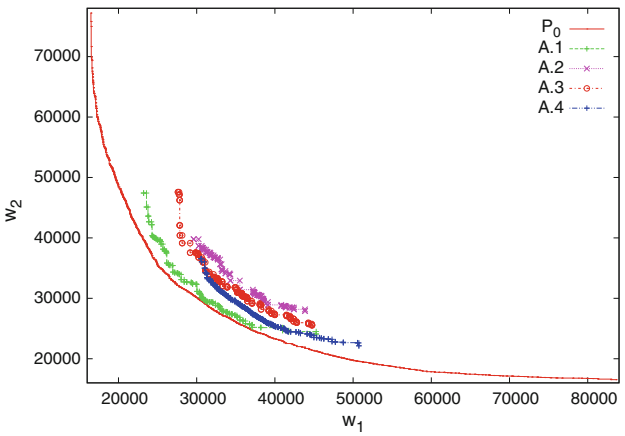

Fig. 1 Example of fronts obtained with the variants of Version 0. Variants could be ordered with relation to their proximity to the reference set as A.1, A.4, A.3, and A.2

3.2 Version 1: multi colony approach

We will first compare the single colony with a multi colony algorithm. In the multi colony case, several colonies cooperate in finding good solutions. This cooperation is achieved by the exchange of information at certain times (Middendorf et al. 2002).

In Version 1 each of the colonies follows the same steps as the colonies described for Version 0. In the next paragraphs, the differences from Version 0 to Version 1 are explained.

3.2.1 Information exchange between the colonies

In the multi-colony versions (Version 1 and following) an exchange of information is mandatory. To do this, at each predefined number of colony cycles, the approximation sets of the $N+1$ colonies, $\mathcal{C}_i^{\mathcal{P}}(i=0,1,\ldots,N)$, are merged into a global one,

$$\mathcal{P} = \mathcal{C}_0^{\mathcal{P}} \uplus \mathcal{C}_1^{\mathcal{P}} \uplus \cdots \uplus \mathcal{C}_N^{\mathcal{P}}. \quad (8)$$

Then the colonies approximation sets are reinitialized with elements of the global approximation set which is split among several colonies, considering the following process:

- Lexicographically sort $\mathcal{P} = \{S_1, S_2, \ldots, S_{|\mathcal{P}|}\}$;
- Colony's 0 approximation set, $\mathcal{C}_0^{\mathcal{P}}$, is reinitialized with the first $\left\lfloor \frac{|\mathcal{P}|}{N} \right\rfloor$ elements of the lexicographically sorted $\mathcal{P}$. Colony 1 approximation set, $\mathcal{C}_1^{\mathcal{P}}$, is reinitialized with the first $\left\lfloor \frac{|\mathcal{P}|}{N} \right\rfloor$ elements of the lexicographically sorted $\mathcal{P} - \mathcal{C}_0^{\mathcal{P}}$. Analogous, Colony 2 approximation set, $\mathcal{C}_2^{\mathcal{P}}$, is reinitialized with the first $\left\lfloor \frac{|\mathcal{P}|}{N} \right\rfloor$ elements of the lexicographically sorted $\mathcal{P} - (\mathcal{C}_0^{\mathcal{P}} \cup \mathcal{C}_1^{\mathcal{P}})$. The process is repeated until $\mathcal{C}_N^{\mathcal{P}}$, which receives the remaining elements.

The objective is to let each colony explore small regions of the search space. This is achieved by using the splitting strategy in conjunction with the pheromone updating formula that uses the colonies approximation sets, $\mathcal{C}_i^{\mathcal{P}}$, to reinitialize the pheromone matrices (Sect. 2.3.2). This procedure is often classified as greedy, in the sense that it only uses the best elements so far, that is, elements of the approximation set.

Furthermore, the splitting of the elements of the approximation set is also motivated by the fact that often the number of elements in that set becomes very large. The observation of the pheromone matrices showed that, if the solutions in the approximation set were all used, then the pheromone-based selection becomes "noisy". In the next sections, we will also notice that this splitting strategy agrees with the settings of the α and β parameters.

The splitting of the approximation set, combined with the pheromone update strategy, is particularly adapted to the multiple objective problems. Firstly because it is supposed that each edge has associated an m-pheromone vector, where each component represents the worthy of that edge in the construction of good solutions, considering a particular weight. Secondly because the approximation set serves as a database of the knowledge acquired by the ants. Nevertheless, it is easy to derive a similar strategy for the single objective case, by providing a list of the best-known solutions.

Algorithm 2 systematizes the multiple colony process.

Algorithm 2 Multiple Colony MOACO Algorithm

```
Initialize the approximation set, P = ∅
for colony_i ∈ {colony_0, colony_1, ..., colony_N} do
    Set parameters for colony_i                          ▷ α_i, β_i, ...
end for
repeat
    for all colony_i ∈ {colony_0, colony_1, ..., colony_N} do
        Use P to set the approximation set of colony_i, C_i^P,
        Update the pheromone matrices of colony_i using C_i^P (see Section 2.3.2)
    end for
    for all colony_i ∈ {colony_0, colony_1, ..., colony_N} do
        repeat
            for all ants ∈ colony_i do
                Compute a solution based on the pheromone trails and possible heuristics of colony_i
                Evaluate the solution and update the approximation set, C_i^P
            end for
            Use C_i^P to update the pheromone trails of colony_i
        until colony_i has made a predefined number of cycles
    end for
    Update P = C_0^P ⊎ C_1^P ⊎ C_N^P
until stopping criteria is met
return Approximation set, P.
```

3.2.2 Comparing Version 0 with Version 1

Table 4 summarizes the combination of parameter values used by the tested variants of Version 1.

The mean and standard deviation statistics for $|P|$,$\mathcal{S}$,$R1_R$, and $R3_R$, are collected in Table 5. Comparing the variants from Version 1, we can observe that the best results were obtained from variants M.1 and M.5. Variant M.1 used 25 colonies against the 50 of M.5, which does not allow us to conclude whichever is the best number of colonies. However, we can observe that the worst cases of the 50 colonies (M.6 and M.7) are slightly better than the worst cases obtained with 25 colonies. Common is the fact that the α 's are equal to 1 and the β's to 1 or 5. $R1_R$ is

Table 4 Parameter values for the tested variants of Versions 1

Variant	Colonies	α	β
M.1	25	1.0	1.0
M.2	25	5.0	1.0
M.3	25	1.0	5.0
M.4	25	5.0	5.0
M.5	50	1.0	5.0
M.6	50	5.0	5.0
M.7	50	5.0	1.0
M.8	50	1.0	1.0

M all variants of this version

Table 5 Version 1 statistics (mean and standard deviation) for the metrics: $|P|$, $\mathcal{S}$, $R1_R$, and $R3_R$

	M.1	M.2	M.3	M.4	M.5	M.6	M.7	M.8
$\|P\|$								
Mean	**1.009e+02**	3.836e+01	7.016e+01	4.100e+01	7.080e+01	6.360e+01	4.420e+01	7.448e+01
SD	4.002e+01	5.773e+00	4.113e+01	5.679e+00	1.027e+01	7.708e+00	6.825e+00	1.045e+01
$\mathcal{S}$								
Mean	**8.736e−01**	7.522e−01	8.563e−01	7.506e−01	8.673e−01	7.854e−01	7.744e−01	8.591e−01
SD	3.631e−02	1.228e−02	5.216e−02	1.901e−02	1.417e−02	1.309e−02	7.023e−03	1.048e−02
$R1_R$								
Mean	**9.998e−01**	1.000e+00	1.000e+00	1.000e+00	1.000e+00	1.000e+00	1.000e+00	1.000e+00
SD	1.010e−03	0.000e+00	0.000e+00	0.000e+00	0.000e+00	0.000e+00	0.000e+00	0.000e+00
$R3_R$								
Mean	−4.320e+03	−8.702e+03	**−3.557e+03**	−7.388e+03	−4.231e+03	−6.797e+03	−8.435e+03	−5.046e+03
SD	1.049e+03	7.469e+02	1.092e+03	1.251e+03	7.381e+02	7.096e+02	6.399e+02	1.068e+03

Best mean value for each case is shown in bold type

totally inconclusive (except for M.1, they are equal to 1 for all variants), showing that the results obtained with variants M.1–M.8 are not close enough to the reference set. $R3_R$ also doesn't allow us to retrieve significant conclusions with all values in the order of -10^3. The cardinality of the approximations sets returned by the variants with $\alpha = 5$ are the lowest, letting us conclude that in those cases, the required exploration of the search space was not achieved.

Compared against the results obtained by Version 0, we can observe that a very slight improvement was achieved, but which cannot be considered sufficient.

Table 6 shows the values of the non-parametric Mann–Whitney paired test (see Appendix B) (Demšar 2006; García et al. 2008, 2009), considering the $\mathcal{S}$ metric results for the independent samples. For the null hypothesis we have considered H_0: "The distribution of $\mathcal{S}(V_i) - \mathcal{S}(V_j)$ is symmetric about 0" against the one-side alternative hypothesis H_1: "$\mathcal{S}(V_i)$ is shifted to the right of $\mathcal{S}(V_j)$ ", where V_i is the variant represented in row i and V_j the one in column j.[4] Compared among the variants of Version 0, we can conclude that A.1 is statistically better than the other three variants and A.4 is better than the other two.

Statistically comparing Versions 0 and 1 (see Table 6), we can observe that variants A.1 and A.4 are better than M.2, M.4, M.6, and M.7. Except for A.1, M.3, and M.5, variant M.1 (from version 1 with 25 colonies) is better than all the others. Except for M.1 and M.3, variant M.5 (with 50 colonies) is better than all the others. Variants M.1, M.3, and M.5 (all from Version 1) are not worse than any of the others. Since the statistic results do not allow us to retain a value for the number of colonies, we will keep both (25 and 50 colonies) in the next versions experiments.

Figure 2 sketches typical fronts obtained from each of the variants (M.1–M.8) against $\mathcal{P}_0$. It is observable that the approximation sets do not present an acceptable quality yet (spread and proximity to the reference set), which justifies the adoption of alternative strategies to be introduced in the next versions.

3.3 Version 2: multiple colonies with different behaviors

This section describes the modifications made between Version 1 and 2. In Version, 2 the behavior of the colonies is different since each colony, i, has different $\alpha^{(i)} = (\alpha_1^{(i)}, \alpha_2^{(i)}, \ldots, \alpha_m^{(i)})$ and $\beta^{(i)} = (\beta_1^{(i)}, \beta_2^{(i)}, \ldots, \beta_m^{(i)})$ parameter values. This is made by setting a maximum value for the α 's, $\alpha_{\max}$, and for the β 's, $\beta_{\max}$. Then the parameters of colony$_i$$(i = 0, 1, \ldots, N)$ are set as:

$$\begin{cases} \alpha_k^{(i)} = \theta_k^{(i)} \times \alpha_{\max} \\ \beta_k^{(i)} = \theta_k^{(i)} \times \beta_{\max} \end{cases} \quad k = 1, 2, \ldots, m \tag{9}$$

where, for $k = 1, 2, \ldots, m$,

$$\theta_k^{(i)} = \max\left\{0, 1 - \left|\frac{m-1}{N} i - (k-1)\right|\right\}. \tag{10}$$

Figure 3 sketches two examples for the θ 's values when (a) $N = 30$ and $m = 2$ weights; and (b) $N = 60$ and $m = 5$ weights. For instance, in the presented examples, the θ 's of

[4] In Table 6, the intersection of row i with column j presents the value of p and the confidence interval at 95% for the Mann–Whitney paired test. When p is small (<0.05) we use symbol ∇ to indicate that we should reject the null hypothesis, that is, we can reject the idea that the observed differences are a coincidence and conclude that the populations have different medians (in our case, $\mathcal{S}(V_i)$ is shifted to the right of $\mathcal{S}(V_j)$). If p is large, marked with symbol ▲, the data does not allow us to conclude that the medians are different, which is not the same as to say that they are equal. For instance, row 1 column 2 has $p = 0.00$ and the confidence interval $(0.1, \infty)$. In this case, since p is small the null hypothesis should be rejected, accepting that $\mathcal{S}(V_1)$ is shifted to the right of $\mathcal{S}(V_2)$. This fact is corroborated by the confidence interval. We can conclude that statistically V_1 has a larger $\mathcal{S}$ value than V_2.

Table 6 Mann–Whitney test (or Wilcoxon rank sum test) results comparing Versions 0 and 1. A.1–A.4 are variants from Version 0 and M.1–M.8 are from Version 1 (separated by the horizontal/vertical lines)

A.1	▽ 0.00 (0.10;∞)	▽ 0.00 (0.08;∞)	▽ 0.00 (0.01;∞)	▲ 0.92 (−0.02;∞)	▽ 0.00 (0.10;∞)	▲ 0.09 (−0.01;∞)	▽ 0.00 (0.10;∞)	▲ 0.96 (−0.01;∞)	▽ 0.00 (0.07;∞)	▽ 0.00 (0.08;∞)	▲ 0.21 (−0.00;∞)
▲ 1.00 (−0.12;∞)	A.2	▲ 1.00 (−0.03;∞)	▲ 1.00 (−0.10;∞)	▲ 1.00 (−0.13;∞)	▲ 0.59 (−0.01;∞)	▲ 1.00 (−0.12;∞)	▲ 0.60 (−0.01;∞)	▲ 1.00 (−0.12;∞)	▲ 1.00 (−0.04;∞)	▲ 1.00 (−0.03;∞)	▲ 1.00 (−0.11;∞)
▲ 1.00 (−0.09;∞)	▽ 0.00 (0.02;∞)	A.3	▲ 1.00 (−0.07;∞)	▲ 1.00 (−0.11;∞)	▽ 0.00 (0.01;∞)	▲ 1.00 (−0.11;∞)	▽ 0.00 (0.01;∞)	▲ 1.00 (−0.10;∞)	▲ 0.99 (−0.02;∞)	▲ 0.61 (−0.01;∞)	▲ 1.00 (−0.09;∞)
▲ 1.00 (−0.04;∞)	▽ 0.00 (0.06;∞)	▽ 0.00 (0.04;∞)	A.4	▲ 1.00 (−0.06;∞)	▽ 0.00 (0.06;∞)	▲ 0.92 (−0.06;∞)	▽ 0.00 (0.06;∞)	▲ 1.00 (−0.05;∞)	▽ 0.00 (0.03;∞)	▽ 0.00 (0.04;∞)	▲ 1.00 (−0.04;∞)
▲ 0.08 (−0.00;∞)	▽ 0.00 (0.11;∞)	▽ 0.00 (0.08;∞)	▽ 0.00 (0.02;∞)	M.1	▽ 0.00 (0.11;∞)	▲ 0.06 (−0.00;∞)	▽ 0.00 (0.11;∞)	▲ 0.40 (−0.01;∞)	▽ 0.00 (0.07;∞)	▽ 0.00 (0.09;∞)	▽ 0.03 (0.00;∞)
▲ 1.00 (−0.12;∞)	▲ 0.42 (−0.01;∞)	▲ 1.00 (−0.03;∞)	▲ 1.00 (−0.10;∞)	▲ 1.00 (−0.13;∞)	M.2	▲ 1.00 (−0.11;∞)	▲ 0.42 (−0.01;∞)	▲ 1.00 (−0.12;∞)	▲ 1.00 (−0.04;∞)	▲ 1.00 (−0.03;∞)	▲ 1.00 (−0.11;∞)
▲ 0.92 (−0.04;∞)	▽ 0.00 (0.07;∞)	▽ 0.00 (0.05;∞)	▲ 0.08 (−0.00;∞)	▲ 0.94 (−0.04;∞)	▽ 0.00 (0.07;∞)	M.3	▽ 0.00 (0.07;∞)	▲ 0.94 (−0.04;∞)	▽ 0.00 (0.04;∞)	▽ 0.00 (0.05;∞)	▲ 0.94 (−0.03;∞)
▲ 1.00 (−0.12;∞)	▲ 0.41 (−0.01;∞)	▲ 1.00 (−0.03;∞)	▲ 1.00 (−0.10;∞)	▲ 1.00 (−0.13;∞)	▲ 0.59 (−0.01;∞)	▲ 1.00 (−0.12;∞)	M.4	▲ 1.00 (−0.12;∞)	▲ 1.00 (−0.04;∞)	▲ 1.00 (−0.03;∞)	▲ 1.00 (−0.12;∞)
▽ 0.04 (0.00;∞)	▽ 0.00 (0.11;∞)	▽ 0.00 (0.09;∞)	▽ 0.00 (0.02;∞)	▲ 0.61 (−0.02;∞)	▽ 0.00 (0.11;∞)	▲ 0.06 (−0.01;∞)	▽ 0.00 (0.11;∞)	M.5	▽ 0.00 (0.08;∞)	▽ 0.00 (0.09;∞)	▽ 0.00 (0.00;∞)
▲ 1.00 (−0.08;∞)	▽ 0.00 (0.03;∞)	▽ 0.02 (0.00;∞)	▲ 1.00 (−0.06;∞)	▲ 1.00 (−0.10;∞)	▽ 0.00 (0.03;∞)	▲ 1.00 (−0.09;∞)	▽ 0.00 (0.03;∞)	▲ 1.00 (−0.09;∞)	M.6	▽ 0.00 (0.01;∞)	▲ 1.00 (−0.08;∞)
▲ 1.00 (−0.09;∞)	▽ 0.00 (0.02;∞)	▲ 0.39 (−0.01;∞)	▲ 1.00 (−0.07;∞)	▲ 1.00 (−0.11;∞)	▽ 0.00 (0.02;∞)	▲ 1.00 (−0.09;∞)	▽ 0.00 (0.02;∞)	▲ 1.00 (−0.10;∞)	▲ 1.00 (−0.02;∞)	M.7	▲ 1.00 (−0.09;∞)
▲ 0.80 (−0.01;∞)	▽ 0.00 (0.10;∞)	▽ 0.00 (0.08;∞)	▽ 0.00 (0.01;∞)	▲ 0.98 (−0.02;∞)	▽ 0.00 (0.10;∞)	▲ 0.06 (−0.01;∞)	▽ 0.00 (0.10;∞)	▲ 1.00 (−0.02;∞)	▽ 0.00 (0.07;∞)	▽ 0.00 (0.08;∞)	M.8

For each pair of variants the Mann–Whitney test p value, the confidence interval, and the symbology ▽|▲ indicating that we should reject the null hypothesis, or the data does not allow us to conclude that the medians are different, respectively, are presented

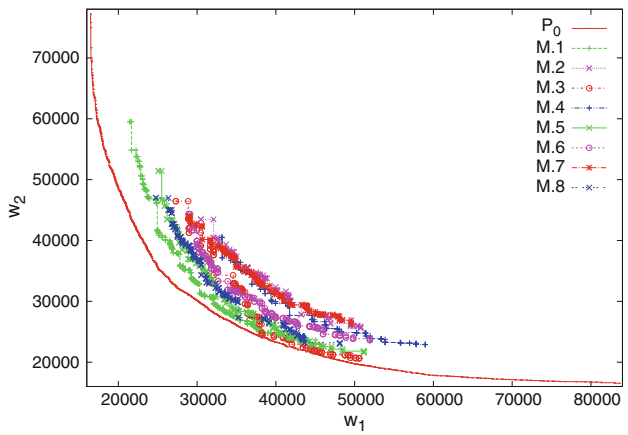

Fig. 2 Example of fronts obtained with the variants of Version 1. The closest variant to the reference set is M.1, followed by M.3, M.5, and M.8

colony ten are equal to (0.67,0.33) and to (0.33,0.67,0,0,0) for $m = 2$ and $m = 5$, respectively.

The θ values, and consequently those of the α's and β's, are intrinsically related with the methodology used to exchange the information between the colonies, as seen in Sect. 3.2.1. For example, the first colony receives the solutions with lower first weights and higher α_1 and β_1 values, having preference for solutions in the same objective area (analogous deductions can be made for the other values).

3.3.1 *Comparing Version 2 with the previous best variants*

In this section, the best previous variants (M.1 and M.5) will be compared with the ones from Version 2 (parameter values are summarized in Table 7).

Table 8 presents the fundamental statistics for the values of the metrics ($|P|$,$\mathcal{S}$,$R1_R$, and $R3_R$) for Version 2 variants. It is observable that: variant 2M.7 ($\alpha_{\max} = \beta_{\max} = 1$ and 50 colonies) obtained the best mean value for the $\mathcal{S}$ metric, followed by 2M.4 and 2M.2. Referring to the mean value of metric $R1$ the best variant was 2M.2 with parameters $\alpha_{\max} = \beta_{\max} = 1$ and 25 colonies. However, the best mean value for metric $R3$ was obtained for variant 2M.8 with $\alpha_{\max} = 5$,$\beta_{\max} = 1$ and 50 colonies. The values of $R3$ do not allow us to discard the cases where $\alpha_{\max} = 5$.

The comparison with the best variants of Versions 0 and 1 allows us to conclude that all metrics values were in general significantly improved. For example, the value of

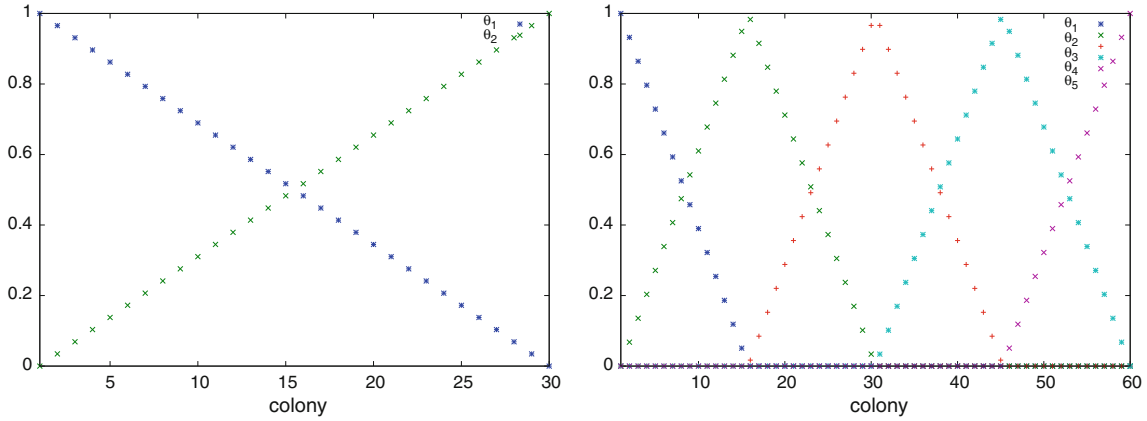

Fig. 3 Example of θ values for **a** $N = 30$ and $m = 2$ weights; and **b** $N = 60$ and $m = 5$ weights

Table 7 Parameter values for the tested variants of Versions 2

Variant	Colonies	α_{max}	β_{max}
2M.1	25	5.0	5.0
2M.2	25	1.0	5.0
2M.3	25	5.0	1.0
2M.4	25	1.0	1.0
2M.5	50	1.0	5.0
2M.6	50	5.0	5.0
2M.7	50	1.0	1.0
2M.8	50	5.0	1.0

2M all variants of this version

the hyper-volume ratio mean in Version 2 is 0.9307 for the worst case, against the best 0.8736 value obtained by the previous versions. Similarly, the higher mean for the cardinality of the approximation set in the previous versions was 100.9, which is less than the smallest of the mean values, obtained for variant 2M.6 (109.2).

Table 9 shows the results for the Mann–Whitney test comparing the variants from Version 2 against themselves and against the best performing variants from the previous versions. According to the values presented in the first two rows and columns of the table, for the $\mathcal{S}$ metric, the variants of Version 2 are significantly better than those of the previous versions, that is, statistically the median hypervolume of Version 2 (all cases) is higher than median of the previously best variants (M.1 and M.5). Referring to Version 2, the median of the $\mathcal{S}$ metric for variants 2M.2, 2M.4, and 2M.7 is significantly larger than for the other ones. Between these last three variants, 2M.4 and 2M.7 have a better performance than 2M.2.

Examples of fronts obtained from each variant of Version 2 are sketched in Fig. 4. The sketched approximations allow us to observe that the cases with higher spread are obtained for variant 2M.1, 2M.3, 2M.6, and 2M.8, with $\alpha_{max} = 5$. It is also observable that, the approximation sets are closer to the reference set and larger spread and diversity were obtained, corroborating the observed metrics values.

Table 8 Version 2 statistics (mean and standard deviation) for the metrics: $|P|$, $\mathcal{S}$, $R1_R$, and $R3_R$

	2M.1	2M.2	2M.3	2M.4	2M.5	2M.6	2M.7	2M.8
$\|P\|$								
Mean	2.550e+02	3.560e+02	1.485e+02	**3.571e+02**	1.601e+02	1.092e+02	2.227e+02	1.423e+02
SD	3.861e+01	3.894e+01	1.328e+01	5.068e+01	5.327e+01	2.564e+01	2.781e+01	9.793e+00
$\mathcal{S}$								
Mean	9.733e−01	9.762e−01	9.389e−01	9.789e−01	9.657e−01	9.307e−01	**9.805e−01**	9.334e−01
SD	3.252e−03	4.132e−03	3.358e−03	4.294e−03	8.492e−03	1.222e−02	2.669e−03	3.213e−03
$R1_R$								
Mean	9.952e−01	**9.388e−01**	9.988e−01	9.711e−01	9.877e−01	9.994e−01	9.964e−01	9.990e−01
SD	5.919e−03	3.185e−02	2.640e−03	2.380e−02	1.459e−02	1.675e−03	6.763e−03	2.062e−03
$R3_R$								
Mean	−2.950e+02	−1.629e+03	−1.971e+02	−1.358e+03	−1.271e+03	−2.832e+02	−6.198e+02	**−1.164e+02**
SD	1.589e+02	4.588e+02	7.107e+01	3.718e+02	5.110e+02	1.655e+02	3.213e+02	5.717e+01

Best mean value for each case is shown in bold type

Table 9 Mann–Whitney test (or Wilcoxon rank sum test) results comparing Versions 2 against the previous best variants

M.1	▲ 0.40 (−0.01;∞)	▲ 1.00 (−0.11;∞)	▲ 1.00 (−0.11;∞)	▲ 1.00 (−0.08;∞)	▲ 1.00 (−0.12;∞)	▲ 1.00 (−0.11;∞)	▲ 1.00 (−0.07;∞)	▲ 1.00 (−0.12;∞)	▲ 1.00 (−0.07;∞)
▲ 0.61 (−0.02;∞)	M.5	▲ 1.00 (−0.11;∞)	▲ 1.00 (−0.11;∞)	▲ 1.00 (−0.07;∞)	▲ 1.00 (−0.11;∞)	▲ 1.00 (−0.10;∞)	▲ 1.00 (−0.07;∞)	▲ 1.00 (−0.12;∞)	▲ 1.00 (−0.07;∞)
▽ 0.00 (0.10;∞)	▽ 0.00 (0.10;∞)	2M.1	▲ 0.99 (−0.00;∞)	▽ 0.00 (0.03;∞)	▲ 1.00 (−0.01;∞)	▽ 0.00 (0.00;∞)	▽ 0.00 (0.04;∞)	▲ 1.00 (−0.01;∞)	▽ 0.00 (0.04;∞)
▽ 0.00 (0.10;∞)	▽ 0.00 (0.10;∞)	▽ 0.01 (0.00;∞)	2M.2	▽ 0.00 (0.04;∞)	▲ 0.98 (−0.00; ∞)	▽ 0.00 (0.01;∞)	▽ 0.00 (0.04;∞)	▲ 1.00 (−0.01;∞)	▽ 0.00 (0.04;∞)
▽ 0.00 (0.06;∞)	▽ 0.00 (0.07;∞)	▲ 1.00 (−0.04;∞)	▲ 1.00 (−0.04;∞)	2M.3	▲ 1.00 (−0.04;∞)	▲ 1.00 (−0.03;∞)	▽ 0.01 (0.00;∞)	▲ 1.00 (−0.04;∞)	▽ 0.00 (0.00;∞)
▽ 0.00 (0.10;∞)	▽ 0.00 (0.11;∞)	▽ 0.00 (0.00;∞)	▽ 0.02 (0.00;∞)	▽ 0.00 (0.04;∞)	2M.4	▽ 0.00 (0.01;∞)	▽ 0.00 (0.04;∞)	▲ 0.84 (−0.00;∞)	▽ 0.00 (0.04;∞)
▽ 0.00 (0.08;∞)	▽ 0.00 (0.09;∞)	▲ 1.00 (−0.01;∞)	▲ 1.00 (−0.01;∞)	▽ 0.00 (0.03;∞)	▲ 1.00 (−0.01;∞)	2M.5	▽ 0.00 (0.03;∞)	▲ 1.00 (−0.02;∞)	▽ 0.00 (0.03;∞)
▽ 0.00 (0.05;∞)	▽ 0.00 (0.06;∞)	▲ 1.00 (−0.05;∞)	▲ 1.00 (−0.05;∞)	▲ 0.99 (−0.01;∞)	▲ 1.00 (−0.05;∞)	▲ 1.00 (−0.04;∞)	2M.6	▲ 1.00 (−0.06;∞)	▲ 0.68 (−0.01;∞)
▽ 0.00 (0.10;∞)	▽ 0.00 (0.11;∞)	▽ 0.00 (0.01;∞)	▽ 0.00 (0.00;∞)	▽ 0.00 (0.04;∞)	▲ 0.16 (−0.00;∞)	▽ 0.00 (0.01;∞)	▽ 0.00 (0.04;∞)	2M.7	▽ 0.00 (0.05;∞)
▽ 0.00 (0.06;∞)	▽ 0.00 (0.06;∞)	▲ 1.00 (−0.04;∞)	▲ 1.00 (−0.04;∞)	▲ 1.00 (−0.01;∞)	▲ 1.00 (−0.05;∞)	▲ 1.00 (−0.04;∞)	▲ 0.32 (−0.00;∞)	▲ 1.00 (−0.05;∞)	2M.8

Variants M.1 and M.5 are from Version 1 (the previously best ones) and 2M.1–2M.8 are from Version 2 (the versions variants are separated by horizontal/vertical lines). For each pair of variants the Mann–Whitney test p value, the confidence interval, and the symbology ∇|▲ indicating that we should reject the null hypothesis, or the data does not allow us to conclude that the medians are different, respectively, are presented

Although the results significantly improved the ones from the previous versions, the next section will be devoted to the introduction of an hybridization which aims to explore the best solutions. The objective is to use them in the exploitation of their neighborhood, avoiding to rebuild all solutions from the beginning.

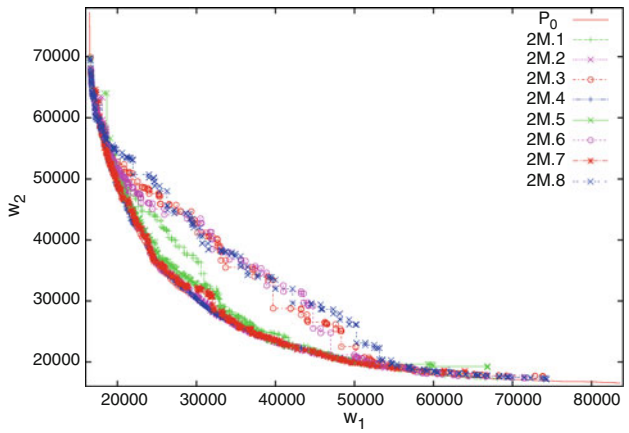

Fig. 4 Example of fronts obtained with Version 2. The closest variants to the reference set are 2M.4, 2M.5, and 2M.7. Variants 2M.1, 2M.3, 2M.6, and 2M.8 (all with $\alpha_{max} = 5$) are far from the reference set in the central region, but present larger diversity

4 The ant colony stochastic depth search exploration

This section presents our proposal of an ant colony stochastic depth search exploration. As we will see, the introduction of new features will significantly improve the previous results.

4.1 Version 3: DANTE

A rudimentary version of an ant colony algorithm builds a solution, evaluates it, performs the pheromone-updating process, and discards it (recall Algorithms 1 and 2). This strategy does not allow a proper local exploration, overlooking the computational effort that was necessary to produce those solutions.

The Depth ANT Explorer (DANTE) process was developed with the objective of further exploiting the constructed solutions. The method is characterized by a restricted depth search based on the pheromone vectors. In other words, a depth search procedure, limited in the number of branches, in the depth level, and/or in the available computational allowance, is performed by each ant (Fig. 5). The depth of the search depends on the solution entering into the approximation set. Further details are given below.

The steps to build a solution were already described for Version 0. As seen, it is a constructive procedure in the sense that, edges are added into the cycle in a given order

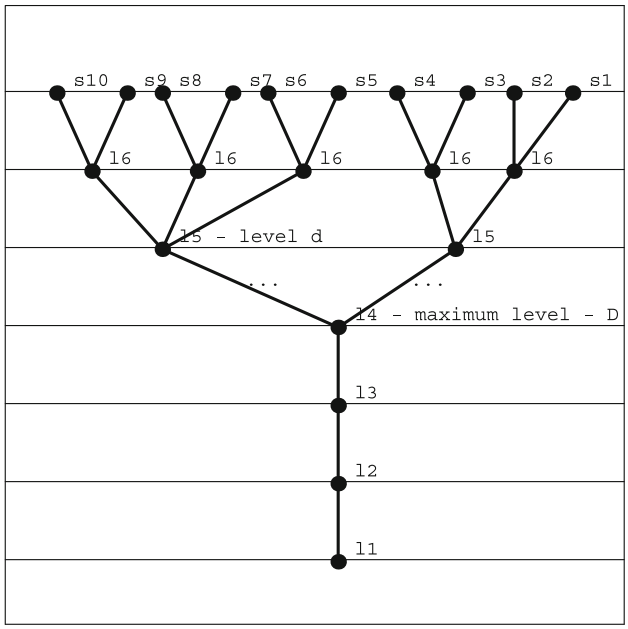

Fig. 5 Limited depth search. $s1$, $s2$,..., sn are the constructed solutions and $l1$, $l2$,... are the depth search levels

until a solution, S, is attained. To better understand the process, imagine a search tree where S is a path from the root (level 0) to one leaf (level n, the number of edges in the path). Then DANTE procedure can be described as follows:

Step 1. Build a solution using the constructive procedure.

Step 2. If solution S does not improve the approximation set then allow M search branches from level d (algorithm parameter) into level n and 0 for the remaining levels:

$$\begin{cases} \text{branches}_l = M & \text{if } d \leq l < n \\ \text{branches}_l = 0 & \text{if } 0 \leq l < d, \end{cases} \tag{11}$$

Step 3. If solution S is inserted into the approximation set $\mathcal{P}$, then allow M search branches from level D (algorithm parameter) into level n and 0 for the remaining levels:

$$\begin{cases} \text{branches}_l = M & \text{if } D \leq l < n \\ \text{branches}_l = 0 & \text{if } 0 \leq l < D \end{cases}. \tag{12}$$

Step 4. From solution S, and from reverse order of its construction, remove edges until reaching a level, l, where new branches are allowed $(\text{branches}_l > 0)$ or $l = 0$.

Step 5. If there are available branches $(l > 0)$ then decrease the number of available branches at level $l(\text{branches}_l - 1)$ and go to Step 6. If there are no available branches at any level $(\text{branches}_l = 0, l = 0, 1, \ldots, n)$ then stop the process.

Step 6. Use the same constructive procedure that was used to build S (as described in Sect. 3.1), to rebuild the solution starting from the path obtained after last step.

Step 7. Try to insert the new solution into the approximation set. If the new solution is inserted into the approximation set then go to Step 3. Otherwise, go to Step 4.

Another difference is the introduction of a taboo list such that the first edge inserted into the solution to be rebuild in Step 3 of Version 3, has not been used in previous rebuilds at that same level of the search tree. This taboo list is reinitialized every time that the rebuilding of the solution is started from a different level of the one of the last rebuild.

The introduction of the taboo list was motivated by the fact that, for the same depth search phase, a large percentage of solutions was repeated by each ant. Therefore, the introduction of the taboo list forces a larger exploitation of the search space, by obligating the algorithm to find other solutions than the previous ones.

Algorithm 3 presents a high-level description of DANTE's method.

Algorithm 3 DANTE Algorithm for a colony

Ensure: Approximation (sub)set, $\mathcal{P}$.
1: Use $\mathcal{P}$ to initialize the pheromone matrices
2: **while** stopping criterion is not met **do**
3: **for all** ants **do**
4: Use the current pheromone trail to build a new solution, $\mathcal{S}$.
5: Try to insert $\mathcal{S}$ in $\mathcal{P}$.
6: **if** the new solution was inserted into $\mathcal{P}$ **then**
7: Allow new search tree branches up to level D
8: **else**
9: Allow new search tree branches up to level d
10: **end if**
11: Use the previous built solution as a base to apply the depth search procedure.
12: **end for**
13: Update the pheromone trail. ▷ *Optional*
14: **end while**

4.1.1 Comparing DANTE with previous versions

In this section DANTE is compared with the best performing variants of the earlier versions. The tested variants parameter values are presented in Table 10, where $|T_l|$ is the size of the taboo list. Statistical values for the used metrics are collected in Table 11, and the Mann–Whitney test values in Table 12.

The analysis of the results allow us to make some observations:

- The best $\mathcal{S}$ mean values were obtained with variants D.5 and D.6, both with 50 colonies and $\alpha_{\max}$ equal to 1; The Mann–Whitney statistical test corroborates the fact that D.5 and D.6 are the best variants.

Table 10 Parameter values for the DANTE variants

Variant	Colonies	$\alpha_{\max}$	$\beta_{\max}$	D	d	M	$\lvert T_l \rvert$
D.1	25	1	1	30	40	5	10
D.2	25	1	5	30	40	5	10
D.3	25	5	1	30	40	5	10
D.4	25	5	5	30	40	5	10
D.5	50	1	1	30	40	5	10
D.6	50	1	5	30	40	5	10
D.7	50	5	1	30	40	5	10
D.8	50	5	5	30	40	5	10

D all variants of this version

- From the previous versions, the best mean value for $\mathcal{S}$ was obtained for 2M.7 which is only better than D.3, D.7 and D.8 (all with $\alpha_{\max} = 5$).
- The second best variants, according to the Mann–Whitney test, are D.1 and D.2, which are still better than the best performing variants from the earlier versions.
- Variant D.2 (with 25 colonies, $\alpha_{\max} = 1$, and $\beta_{\max} = 5$) obtained the best values for $|P|$ and $R1_R$;
- The best $R3_R$ value was obtained with variant D.7 (with $\alpha_{\max} = 5$ and $\beta_{\max} = 1$).

According to the Mann–Whitney test we can accept that variants D.5 and D.6 (from DANTE version) have a higher $\mathcal{S}$ median value, compared to all the previous ones. Since we could not discard cases where $\alpha_{\max} = 5$, it was decided to consider the combination of the best performing parameters variants, as we will see in the next version.

Figure 6 sketches typical fronts for the DANTE's variants. The sketched approximations allow us to observe that variants D.3, D.4, and D.7 (all with $\alpha_{\max} = 5$) are the ones with higher diversity, and the closest variants to the reference set are D.2, D.5, and D.6.

4.2 Version 4: DANTE II

Version 4 can be considered as a subversion of DANTE. For this version two modifications were introduced:

- First two types of colonies were considered: (a) colonies that behave (i.e., have $\alpha_{\max}$ and $\beta_{\max}$ parameters defined) as the ones described in Sect. 3.3, and (b) colonies where the $\alpha'_{\max}$ and $\beta'_{\max}$ parameters are defined in the reverse way, that is, $\alpha'_{\max} = \beta_{\max}$ and $\beta'_{\max} = \alpha_{\max}$. This strategy is justified by the observation that, according to different metrics, in some cases it was better to have larger α's, while for other it was better to have larger β 's.
- Second we decided to limit the maximum number of solutions that each ant can build within the depth search procedure. This modification tries to limit the increased computation required by the use of the two types of colonies. It was observed that, for the experimented time, some variants produced a rather small number of cycles, which leads to a poor information exchange between the colonies.

4.2.1 *Comparing Version 4 (DANTE II) with the previous ones*

To test the proposed changes $\alpha_{\max}$ was set equal to 5 and $\beta_{\max}$ to 1 for the first type of colonies and the reverse ($\alpha'_{\max} = 1$ and $\beta'_{\max} = 5$) for second set of colonies. The remaining parameter values are summarized in Table 13, where N_S is the maximum number of solutions which an ant is allowed to build, and "Colonies" is the number of colonies for each of the two types.

Table 11 Version 3 (DANTE) statistics (mean and standard deviation) for the metrics: $|P|$, $\mathcal{S}$, $R1_R$, and $R3_R$

	D.1	D.2	D.3	D.4	D.5	D.6	D.7	D.8
$\lvert P \rvert$								
Mean	4.935e+02	**5.833e+02**	2.224e+02	3.563e+02	3.877e+02	4.911e+02	2.072e+02	2.906e+02
SD	3.943e+01	3.199e+01	1.783e+01	3.279e+01	3.458e+01	3.506e+01	1.441e+01	2.341e+01
$\mathcal{S}$								
Mean	9.889e−01	9.879e−01	9.632e−01	9.867e−01	**9.906e−01**	9.900e−01	9.583e−01	9.796e−01
SD	1.915e−03	2.297e−03	5.009e−03	3.001e−03	2.062e−03	1.957e−03	4.600e−03	3.892e−03
$R1_R$								
Mean	8.766e−01	**7.343e−01**	9.853e−01	9.255e−01	9.422e−01	8.335e−01	9.883e−01	9.648e−01
SD	3.982e−02	2.941e−02	3.835e−03	1.936e−02	2.378e−02	2.729e−02	4.051e−03	1.076e−02
$R3_R$								
Mean	−3.664e+02	−5.869e+02	−6.078e+00	−1.280e+01	−2.273e+02	−4.479e+02	**−4.096e+00**	−5.532e+00
SD	1.418e+02	2.179e+02	1.664e+01	1.831e+01	1.503e+02	1.946e+02	5.076e+00	9.078e+00

Best mean value for each case is shown in bold type

Table 12 Mann–Whitney test (or Wilcoxon rank sum test) results comparing Version 3 (DANTE) against the previous best variants (the versions variants are separated by horizontal/vertical lines)

2M.4	▲ 0.84 (−0.00;∞)	▲ 1.00 (−0.01;∞)	▲ 1.00 (−0.01;∞)	▽ 0.00 (0.01;∞)	▲ 1.00 (−0.01;∞)	▲ 1.00 (−0.01;∞)	▲ 1.00 (−0.01;∞)	▽ 0.00 (0.02;∞)	▲ 0.64 (−0.00;∞)
▲ 0.16 (−0.00;∞)	2M.7	▲ 1.00 (−0.01;∞)	▲ 1.00 (−0.01;∞)	▽ 0.00 (0.02;∞)	▲ 1.00 (−0.01;∞)	▲ 1.00 (−0.01;∞)	▲ 1.00 (−0.01;∞)	▽ 0.00 (0.02;∞)	▲ 0.19 (−0.00;∞)
▽ 0.00 (0.01;∞)	▽ 0.00 (0.01;∞)	D.1	▲ 0.14 (−0.00;∞)	▽ 0.00 (0.02;∞)	▽ 0.00 (0.00;∞)	▲ 1.00 (−0.00;∞)	▲ 0.98 (−0.00;∞)	▽ 0.00 (0.03;∞)	▽ 0.00 (0.01;∞)
▽ 0.00 (0.01;∞)	▽ 0.00 (0.01;∞)	▲ 0.87 (−0.00;∞)	D.2	▽ 0.00 (0.02;∞)	▽ 0.05 (0.00;∞)	▲ 1.00 (−0.00;∞)	▲ 1.00 (−0.00;∞)	▽ 0.00 (0.03;∞)	▽ 0.00 (0.01;∞)
▲ 1.00 (−0.02;∞)	▲ 1.00 (−0.02;∞)	▲ 1.00 (−0.03;∞)	▲ 1.00 (−0.03;∞)	D.3	▲ 1.00 (−0.03;∞)	▲ 1.00 (−0.03;∞)	▲ 1.00 (−0.03;∞)	▽ 0.00 (0.00;∞)	▲ 1.00 (−0.02;∞)
▽ 0.00 (0.01;∞)	▽ 0.00 (0.00;∞)	▲ 1.00 (−0.00;∞)	▲ 0.95 (−0.00;∞)	▽ 0.00 (0.02;∞)	D.4	▲ 1.00 (−0.01;∞)	▲ 1.00 (−0.00;∞)	▽ 0.00 (0.03;∞)	▽ 0.00 (0.01;∞)
▽ 0.00 (0.01;∞)	▽ 0.00 (0.01;∞)	▽ 0.00 (0.00;∞)	▽ 0.00 (0.00;∞)	▽ 0.00 (0.03;∞)	▽ 0.00 (0.00;∞)	D.5	▲ 0.20 (−0.00;∞)	▽ 0.00 (0.03;∞)	▽ 0.00 (0.01;∞)
▽ 0.00 (0.01;∞)	▽ 0.00 (0.01;∞)	▽ 0.02 (0.00;∞)	▽ 0.00 (0.00;∞)	▽ 0.00 (0.03;∞)	▽ 0.00 (0.00;∞)	▲ 0.81 (−0.00;∞)	D.6	▽ 0.00 (0.03;∞)	▽ 0.00 (0.01;∞)
▲ 1.00 (−0.02;∞)	▲ 1.00 (−0.02;∞)	▲ 1.00 (−0.03;∞)	▲ 1.00 (−0.03;∞)	▲ 1.00 (−0.01;∞)	▲ 1.00 (−0.03;∞)	▲ 1.00 (−0.03;∞)	▲ 1.00 (−0.03;∞)	D.7	▲ 1.00 (−0.02;∞)
▲ 0.36 (−0.00;∞)	▲ 0.81 (−0.00;∞)	▲ 1.00 (−0.01;∞)	▲ 1.00 (−0.01;∞)	▽ 0.00 (0.01;∞)	▲ 1.00 (−0.01;∞)	▲ 1.00 (−0.01;∞)	▲ 1.00 (−0.01;∞)	▽ 0.00 (0.02;∞)	D.8

For each pair of variants the Mann–Whitney test p value, the confidence interval, and the symbology ∇|▲ indicating that we should reject the null hypothesis, or the data does not allow us to conclude that the medians are different, respectively, are presented

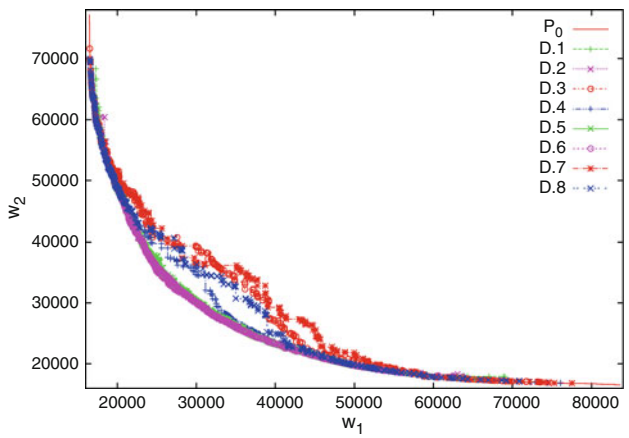

Fig. 6 Typical fronts from the DANTE's variants. Variants with higher diversity are D.3, D.4, and D.7 (all with $\alpha_{max} = 5$). The closest variants to the reference set are D.2, D.5, and D.6

From the analysis of the results (Tables 14 and 15) some conclusions can be made:

- Referring to value of the $\mathcal{S}$ metric, the best variant was 2D.3 with a mean value equal to 0.9960, which is better than the previous best values. This variant used 25 colonies and $N_S = 10$.
- Furthermore, variants 2D.1, 2D.3, and 2D.4, have a better hyper-volume ratio mean than any of the previous versions (the best value was 0.9906 for D.5).
- The best $R1_R$ and $R3_R$ values were obtained by variants 2D.3 and 2D.1, respectively. For both cases, these values are better than those observed in the earlier versions.
- Statistically speaking, the Mann–Whitney test allow us to conclude that variants D.5 and D.6 (the variants with better performance for the previous versions) have a median $\mathcal{S}$ value smaller than that of 2D.1, 2D.3, and 2D.4.
- Up to now, variant 2D.3 was the best variant overall.

The global analysis of the results allow us to conclude that the introduced changes were beneficial for the algorithm performance. A final improvement will be presented in the next section. This final change includes a more selective criterion to enter the oriented depth search, allowing only the best performing solutions to do so.

4.3 Version 5: ϵ-DANTE

The insertion of a solution into the approximation set was the criterion used in DANTE to apply the depth-search phase into the deepest level. Furthermore, the depth search is applied even to the worst solution, although up to a higher level (level d —Eq. 11). Those rules equally exploit solutions which:

Table 13 Parameter values for the DANTE variants

Variant	Colonies	D	d	M	$\lvert T_l \rvert$	N_S
2D.1	25	30	40	5	10	∞
2D.2	50	30	40	5	10	∞
2D.3	25	30	40	5	10	50
2D.4	50	30	40	5	10	50

2D all variants of this version

Table 14 Version 4 (DANTE II) statistics (mean and standard deviation) for the metrics: $\lvert P \rvert$, $\mathcal{S}$, $R1_R$, and $R3_R$

	2D.1	2D.2	2D.3	2D.4
$\lvert P \rvert$				
Mean	3.208e+02	2.473e+02	**5.215e+02**	3.379e+02
SD	2.688e+01	1.626e+01	4.550e+01	5.130e+01
$\mathcal{S}$				
Mean	9.937e−01	9.873e−01	**9.960e−01**	9.928e−01
SD	8.280e−04	1.318e−03	7.179e−04	1.909e−03
$R1_R$				
Mean	9.669e−01	9.962e−01	**8.461e−01**	9.622e−01
SD	1.656e−02	3.934e−03	3.031e−02	2.436e−02
$R3_R$				
Mean	**−1.954e+01**	−2.447e+01	−6.576e+01	−3.777e+01
SD	1.961e+01	2.837e+01	4.568e+01	3.364e+01

Best mean value for each case is shown in bold type

Table 15 Mann–Whitney test (or Wilcoxon rank sum test) results comparing DANTE II against the previous best variants (separated by the horizontal/vertical lines)

D.5	▲ 0.20 (0.00;∞)	▲ 1.00 (0.00;∞)	▽ 0.00 (0.00;∞)	▲ 1.00 (− 0.01;∞)	▲ 1.00 (0.00;∞)
▲ 0.81 (0.00;∞)	D.6	▲ 1.00 (0.00;∞)	▽ 0.00 (0.00;∞)	▲ 1.00 (− 0.01;∞)	▲ 1.00 (0.00;∞)
▽ 0.00 (0.00;∞)	▽ 0.00 (0.00;∞)	2D.1	▽ 0.00 (0.01;∞)	▲ 1.00 (0.00;∞)	▽ 0.02 (0.00;∞)
▲ 1.00 (0.00;∞)	▲ 1.00 (0.00;∞)	▲ 1.00 (− 0.01;∞)	2D.2	▲ 1.00 (− 0.01;∞)	▲ 1.00 (− 0.01;∞)
▽ 0.00 (0.00;∞)	▽ 0.00 (0.01;∞)	▽ 0.00 (0.00;∞)	▽ 0.00 (0.01;∞)	2D.3	▽ 0.00 (0.00;∞)
▽ 0.00 (0.00;∞)	▽ 0.00 (0.00;∞)	▲ 0.98 (0.00;∞)	▽ 0.00 (0.00;∞)	▲ 1.00 (0.00;∞)	2D.4

For each pair of variants the Mann–Whitney test p value, the confidence interval, and the symbology ▽|▲ indicating that we should reject the null hypothesis, or the data does not allow us to conclude that the medians are different, respectively, are presented

- Have objective values equal to elements of the approximation set—solutions possibly built in a different order, or simply distinct solutions but with the same weights;

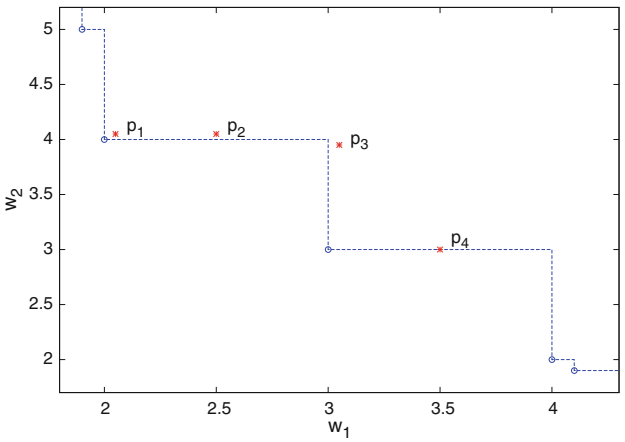

Fig. 7 Points near approximation set front (p_1 is near an element of the approximation set, p_2 and p_3 are near the approximation front, and p_4 is on the approximation front)

- Are very near to the approximation set front—solutions which are almost as good as the ones in the approximation set (Fig. 7); and
- Are very far from the best known solutions—these solutions obtain the same attention as those which are "very near to belong" to the approximation set.

Based on the described ideas, Version 5 introduces a final improvement:

- If a built solution improves the approximation set or its distance to that set is not superior to an ϵ criterion parameter, then the restricted depth search method is applied. In the former case the depth search is applied up to level D, while in the latter it is applied up to level d.
- Solutions which do not satisfy any of the previous criteria are discarded.

The new parameter, ϵ, is introduced with the objective of having a more accurate control of the exploration versus exploitation phases. Larger values of ϵ will imply a more extensive exploitation of the search space, since the depth search is applied to a superior number of solutions. On the other hand, smaller values of this parameter implement the exploration of the search space, since only the best fitting solutions serve as base to the depth search procedure, allowing the allocation of more computational resources to the seeking of new promising areas.

Based on the ϵ range value, this version was called ϵ-DANTE. Putting it all together the ϵ-DANTE process can be summarized as follows.

In ϵ-DANTE at each iteration a colony receives a set of approximations (see Sect. 3.2.1). That set of approximations is used to initialize the pheromone matrices and the

colonies approximation base. Then each agent builds a solution using the pheromone trails and other heuristics. The fitness of the computed solution, S, is compared with the fitness of the elements of the colony's approximation set, $\mathcal{C}_i^{\mathcal{P}}$. If the solution is "far" from the approximation set then it is discarded. Otherwise, according to the quality of these solutions, a limited depth search is made up to two possible levels:

Level **D**: If S improves $\mathcal{C}_i^{\mathcal{P}}$, that is, S is not dominated by any element of $\mathcal{C}_i^{\mathcal{P}}(\nexists_{T \in \mathcal{C}_i^{\mathcal{P}}} : T \prec S)$; or
Level **d**: If S is weakly dominated by some element T of $\mathcal{C}_i^{\mathcal{P}}$ but its relative distance to the approximation set front is smaller than ϵ, which includes the case where the weight vector of S is equal to the weight vector of some solution in $\mathcal{C}_i^{\mathcal{P}}$ that is, $(\mathcal{W}(T) = \mathcal{W}(S))$.

Algorithm 4 presents a high-level description of the ϵ-DANTE method.

Algorithm 4 ϵ-DANTE Algorithm for a colony

Require: $\mathcal{C}_i^{\mathcal{P}}$
Ensure: Approximation set, $\mathcal{P}$.
1: Initialize the pheromone matrices using the approximation set $\mathcal{C}_i^{\mathcal{P}}$
2: **while** stopping criterion is not met **do**
3: **for all** ants **do**
4: Use the current pheromone trail to build a new solution, $\mathcal{S}$.
5: Try to insert $\mathcal{S}$ in $\mathcal{P}$
6: **if** the new solution was inserted into $\mathcal{P}$ **then**
7: Use the previous built solution, $\mathcal{S}$, as a base to apply the depth search procedure up-to branch level D
8: **else**
9: **if** the distance of $\mathcal{S}$ to the approximation set front, $\mathcal{P}$, is inferior to ϵ **then**
10: Use the previous built solution, $\mathcal{S}$, as a base to apply the depth search procedure up-to branch level d
11: **end if**
12: **end if**
13: **end for**
14: Update the pheromone trail. ▷ *Optional*
15: **end while**

4.3.1 *Comparing ϵ-DANTE with the best performing variants*

In this section we compare ϵ-DANTE with the best variants of the previous versions. Table 16 presents the parameter values for the new versions. The values of the α 's and β 's were set as in DANTE II , that is, $\alpha_{\max} = 5$ and $\beta_{\max} = 1$ for the first set of colonies and $\alpha_{\max} = 1$ and $\beta_{\max} = 5$ for second set of colonies. A series of experiments were done varying the ϵ over a set of values ($\{0, 0.01, 0.1\}$). A value for ϵ which gives good general performance was found to be 0.01.

Table 16 Parameter values for the ϵ-DANTE variants

Variant	Colonies	D	d	M	$\lvert T_l \rvert$	ϵ	N_S
ϵD.1	25	30	40	5	10	0.01	∞
εD.2	50	30	40	5	10	0.01	∞
ϵD.3	25	30	40	5	10	0.01	50
ϵD.4	50	30	40	5	10	0.01	50

ϵD all variants of this version

Table 17 Version 5 (ϵ-DANTE) statistics (mean and standard deviation) for the metrics: $\lvert P \rvert$, $\mathcal{S}, R1_R$, and $R3_R$

	ϵD.1	ϵD.2	ϵD.3	ϵD.4
$\lvert P \rvert$				
Mean	**5.911e+02**	4.446e+02	3.143e+02	2.340e+02
SD	3.910e+01	3.655e+01	3.108e+01	2.031e+01
$\mathcal{S}$				
Mean	**9.964e−01**	9.957e−01	9.931e−01	9.861e−01
SD	1.069e−03	7.603e−04	1.256e−03	1.157e−03
$R1_R$				
Mean	**7.962e−01**	9.085e−01	9.766e−01	9.943e−01
SD	3.665e−02	2.737e−02	1.514e−02	4.684e−03
$R3_R$				
Mean	−7.025e+01	−2.434e+01	**−2.353e+01**	−3.458e+01
SD	6.290e+01	2.443e+01	2.951e+01	4.334e+01

Best mean value for each case is shown in bold type

Table 17 summarizes the statistics for the cardinality, $\mathcal{S}$,$R1_R$, and $R3_R$ metrics. Table 18 completes the study with the Mann–Whitney test results. From the analysis of those tables some conclusions can be made:

- Except for $R3_R$,ϵD.1 is the best variant among the ϵDANTE ones. This variant has 50 colonies (25 of each type) and does not limit the number of solutions to be built by each ant ($N_S = \infty$).
- Again except for $R3_R$,ϵD.2 was the second best variant. ϵD.2 has 100 colonies (50 for each type) and it does not also limit the number of solutions to be built by each ant.
- Statistically, the Mann–Whitney test allows us to infer that ϵD.1 was the best variant (referring to $\mathcal{S}$ mean value). Furthermore, the same statistical test allows us to say that ϵD.2 was better than ϵD.3, which was better than ϵD.4. Comparing with the previous version, ϵD.1 improves all variants and ϵD.2 is only comparable to 2D.3.

An overall analysis of the results is made in the next section, where we will use the C coverage metric to compare the best performing variants (with respect to the previously used metric).

Table 18 Mann–Whitney test (or Wilcoxon rank sum test) results comparing ϵ-DANTE against the previous best variants

2D.1	▽ 0.00 (0.01;∞)	▲ 1.00 (− 0.00;∞)	▽ 0.02 (0.00;∞)	▲ 1.00 (− 0.00;∞)	▲ 1.00 (− 0.00;∞)	▲ 0.07 (− 0.00;∞)	▽ 0.00 (0.01;∞)
▲ 1.00 (− 0.01;∞)	2D.2	▲ 1.00 (− 0.01;∞)	▲ 1.00 (− 0.01;∞)	▲ 1.00 (− 0.01;∞)	▲ 1.00 (− 0.01;∞)	▲ 1.00 (− 0.01;∞)	▽ 0.00 (0.00;∞)
▽ 0.00 (0.00;∞)	▽ 0.00 (0.01;∞)	2D.3	▽ 0.00 (0.00;∞)	▲ 0.97 (− 0.00;∞)	▲ 0.15 (− 0.00;∞)	▽ 0.00 (0.00;∞)	▽ 0.00 (0.01;∞)
▲ 0.98 (− 0.00;∞)	▽ 0.00 (0.00;∞)	▲ 1.00 (− 0.00;∞)	2D.4	▲ 1.00 (− 0.00;∞)	▲ 1.00 (− 0.00;∞)	▲ 0.75 (− 0.00;∞)	▽ 0.00 (0.01;∞)
▽ 0.00 (0.00;∞)	▽ 0.00 (0.01;∞)	▽ 0.04 (0.00;∞)	▽ 0.00 (0.00;∞)	ϵ D.1	▽ 0.00 (0.00;∞)	▽ 0.00 (0.00;∞)	▽ 0.00 (0.01;∞)
▽ 0.00 (0.00;∞)	▽ 0.00 (0.01;∞)	▲ 0.85 (− 0.00;∞)	▽ 0.00 (0.00;∞)	▲ 1.00 (− 0.00;∞)	ϵD.2	▽ 0.00 (0.00;∞)	▽ 0.00 (0.01;∞)
▲ 0.93 (− 0.00;∞)	▽ 0.00 (0.01;∞)	▲ 1.00 (− 0.00;∞)	▲ 0.26 (− 0.00;∞)	▲ 1.00 (− 0.00;∞)	▲ 1.00 (− 0.00;∞)	ϵD.3	▽ 0.00 (0.01;∞)
▲ 1.00 (− 0.01;∞)	▲ 1.00 (− 0.00;∞)	▲ 1.00 (− 0.01;∞)	▲ 1.00 (− 0.01;∞)	▲ 1.00 (− 0.01;∞)	▲ 1.00 (− 0.01;∞)	▲ 1.00 (− 0.01;∞)	ϵD.4

2D.1–2D.4 are variants from DANTE II and ϵD.1–ϵD.4 from ϵ-DANTE (separated by the horizontal/vertical lines). For each pair of variants the Mann–Whitney test p value, the confidence interval, and the symbology ▽|▲ indicating that we should reject the null hypothesis, or the data does not allow us to conclude that the medians are different, respectively, are presented

4.4 Overall analysis

In summary, six ACO versions were presented in Sects. 3 and 4. Ranging from Version 0, which can be seen as a classical multiple objective ACO, to our final version, that presents a limited stochastic depth search, ϵ-DANTE. Figure 8 shows a diagram with the proposed evolution between the different versions.

In the remaining of this section, the best performing variants of versions DANTE (I and II) and ϵ-DANTE (D.2, D.5, D.7, 2D.1, 2D.3, ϵD.1, and ϵD.3), in at least one of the studied metrics ($\mathcal{S}$,$R1$, and $R3$), will be compared using the C-coverage metric (Appendix A). We notice that the best variants of DANTE II and ϵ-DANTE (2D.1, 2D.3, ϵD.1, and ϵD.3) used the smallest number of colonies ($50 = 2 \times 25$). This case will be explored in the next section with the introduction of complementary tests with a smaller number of colonies.

Figure 9 sketches the beanplots[5] of the C coverage metric distribution, used to compare the best performing variants among themselves, according to the explained criterion for running times of 300 and 1,800 s (Table 19 summarizes the mean and standard deviation statistics). For example, in Fig. 9 the image in the intersection of row 1 with column 2 represents the beanplot for the C distribution of D.2 against D.5, that is, the percentage of elements of D.5 which are weakly dominated by elements of D.2. The intersection of row 2 with column 1 presents the reverse percentage, that is, the percentage of elements of D.2 which are dominated by elements of D.5. Similar observations can be made for Table 19.

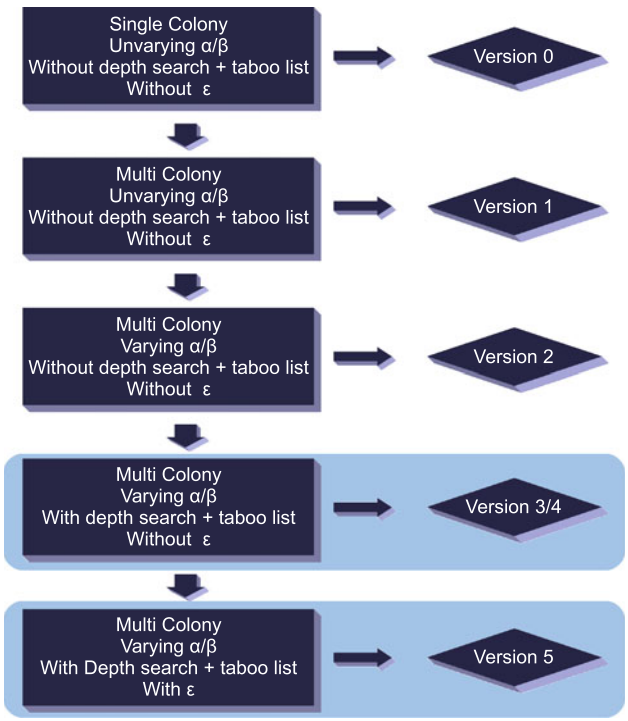

Fig. 8 Diagram of the versions evolution and their main characteristics

[5] A beanplot (Kampstra 2008) combines an 1d-scatter plot and a density trace, being an alternative to boxplots. In a beanplot, the individual observations are shown as small (white color) lines in a one-dimensional scatter plot. Next to that, the estimated density of the distributions is visible and the average is shown (black color region).

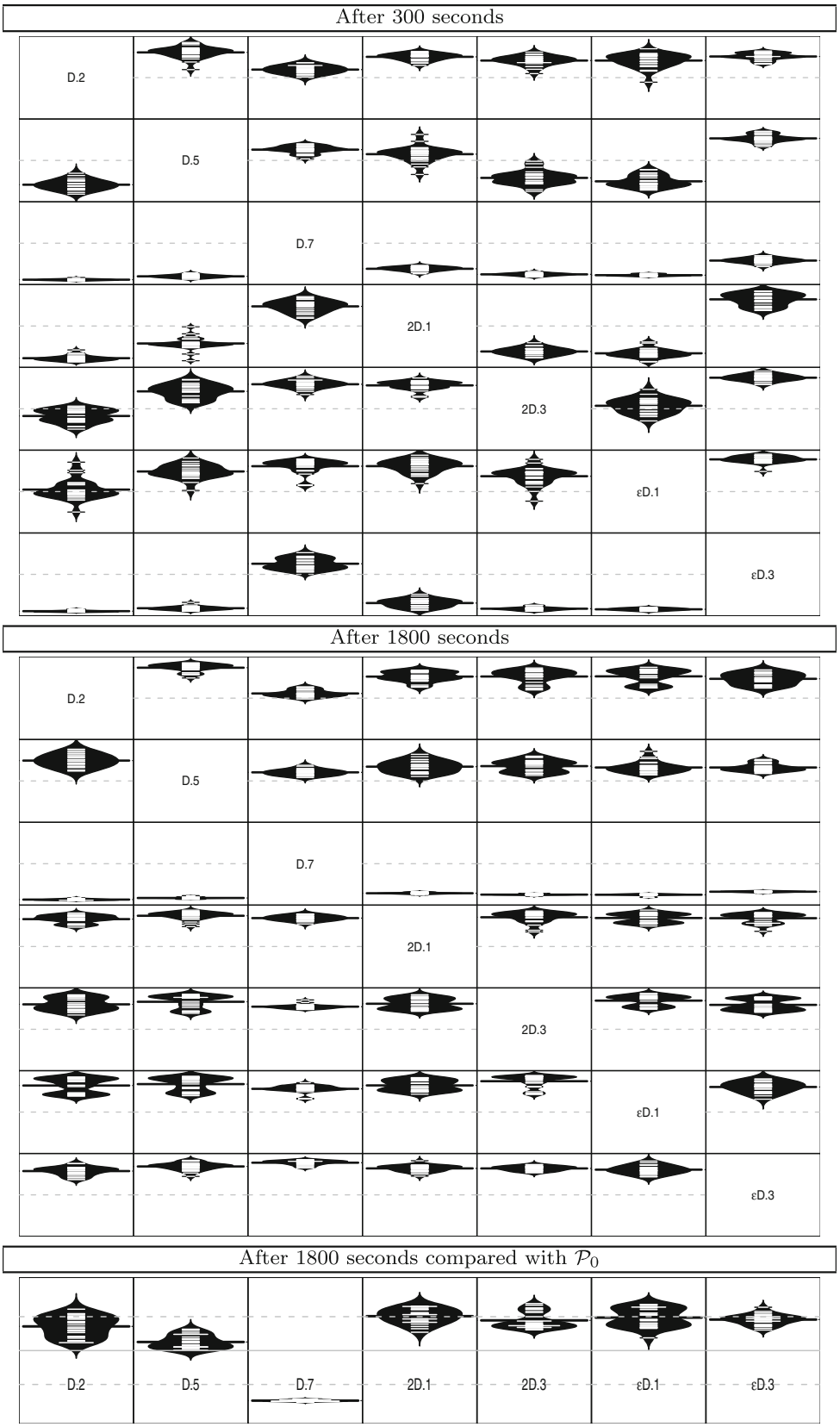

Fig. 9 Beanplots for the *C* metric values, comparing the best performing variants (after 300 and 1,800 s)

Recalling that these results are from a single problem, some conclusions can be made from the analysis of the results:

- The best variants were D.2, 2D.3, and ϵD.1. Each one from one of the versions: DANTE, DANTE II , and ϵ-DANTE. A common point is the fact that they all have the smaller number of colonies: 25 for DANTE and 50 for DANTE II and ϵ-DANTE (from DANTE II it was decided to use two types of colonies).
- The worst variants were D.7, ϵD.3, and 2D.1. D.7 (from DANTE) is worse than all the others, and ϵD.3 (ϵ-DANTE) was only better than D.7. 2D.1 (DANTE II) was better than D.7 and ϵD.3, but worse than all the others. Again, the variants belong to distinct versions of the proposed methods.

For 300 s run time, D.2 was slightly better than 2D.3 and ϵD.1. Although the insertion of the taboo list intended to implement a larger exploration, it was observed that the number of iterations, namely in ϵD.1, was small (about 5). If we analyze the C metric for 1,800 s, the results are slightly different:

- D.7 (DANTE) remains as the worst variant overall.
- All variants, except for D.7, have a more alike behavior. Nevertheless, D.5 (DANTE) is slightly worse than the remaining ones.
- 2D.1 (DANTE II) was better than all the others, that is, the percentage of elements of the other variants dominated by the elements of 2D.1 is larger than the reverse percentage $(C(2D.1, \_) > C(\_, 2D.1))$. Nevertheless, the differences from ϵD.1 and ϵD.3 (both from ϵ-DANTE) are as small as 2% (0.87–0.85).

For the 1,800 s the best variants were from versions DANTE II and ϵ-DANTE. This seems to indicate that, given enough computational resources, the use of two types of colonies produces a better exploration/exploitation of the search space.

The results of the comparison of C metric values against the reference set are also collected in Table 19 and Fig. 9. Important is the fact that between 72 and 75% of the solutions obtained by variants 2D.1 (DANTE II), 2D.3 (DANTE II), ϵD.1 (ϵ-DANTE), and ϵD.3 (ϵ-DANTE) belong to the reference set.

In conclusion, a great evolution of the solutions quality is attained from the final versions when compared with the first classical algorithmic solutions (which we can consider as Versions 0 to 2). The metrics reported these evolutions which were corroborated by the applied statistical tests. Nevertheless, the study presented in this section was made for a single problem instance. Therefore, the next section will be dedicated to the comparison of the best performing of the proposed methods over a larger set of instances, including some with higher dimensions.

5 Complementary experimental results

To complete the study, this section presents the comparisons between the best performing variants proposed in the

Table 19 C metric statistics (mean and standard deviation) comparing the best performing variants between them (after 300 and 1,800 s) and against the reference set

After 300 s						
D.2	0.83 (0.08)	0.60 (0.05)	0.77 (0.05)	0.72 (0.06)	0.72 (0.09)	0.78 (0.05)
0.18 (0.07)	D.5	0.64 (0.05)	0.58 (0.10)	0.27 (0.09)	0.23 (0.08)	0.79 (0.05)
0.02 (0.01)	0.07 (0.02)	D.7	0.17 (0.03)	0.09 (0.02)	0.08 (0.01)	0.27 (0.03)
0.08 (0.04)	0.27 (0.08)	0.76 (0.08)	2D.1	0.17 (0.05)	0.14 (0.06)	0.85(0.08)
0.41 (0.09)	0.73 (0.09)	0.82 (0.05)	0.81 (0.06)	2D.3	0.54 (0.10)	0.91 (0.04)
0.53 (0.13)	0.76 (0.09)	0.83 (0.09)	0.83 (0.09)	0.70 (0.11)	ϵD.1	0.92 (0.05)
0.02 (0.01)	0.06 (0.02)	0.64 (0.07)	0.13 (0.05)	0.05 (0.02)	0.04 (0.01)	ϵD.3
After 1,800 s						
D.2	0.90 (0.05)	0.56 (0.04)	0.78 (0.07)	0.78 (0.09)	0.78 (0.09)	0.75 (0.08)
0.77 (0.08)	D.5	0.61 (0.04)	0.69 (0.08)	0.69 (0.08)	0.68 (0.07)	0.68 (0.05)
0.03 (0.01)	0.05 (0.01)	D.7	0.11 (0.01)	0.10 (0.01)	0.09 (0.01)	0.13 (0.01)
0.86 (0.05)	0.90 (0.05)	0.87 (0.03)	2D.1	0.88 (0.07)	0.87 (0.06)	0.87 (0.06)
0.83 (0.08)	0.86 (0.09)	0.80 (0.03)	0.83 (0.07)	2D.3	0.87 (0.07)	0.82 (0.07)
0.85 (0.10)	0.86 (0.09)	0.80 (0.05)	0.85 (0.07)	0.90 (0.08)	ϵD.1	0.83 (0.07)
0.81 (0.06)	0.87 (0.05)	0.92 (0.03)	0.85 (0.04)	0.84 (0.03)	0.83 (0.05)	ϵD.3
After 1,800 s compared with $\mathcal{P}_0$						
D.2	D.5	D.7	2D.1	2D.3	ϵD.1	ϵD.3
0.68 (0.08)	0.56 (0.05)	0.13 (0.01)	0.75 (0.05)	0.72 (0.06)	0.74 (0.07)	0.73 (0.04)

previous section. To compare between the best performing methods proposed in the previous section, a total of six additional problems were tested: three with two objectives (*kroAC50, kroAD50*, and *kroAE50*), one with three objectives (*kroABC50*), one with four objectives (*kroABCD50*), and one with 100 nodes and two objectives (*kroAB100*). For each problem, the selected methods were run 25 times with common parameter values summarized in Table 20.

5.1 Problems with 50 nodes and 2 objectives

Table 21 summarizes the parameter values used to test the 50 nodes and two objective cases (kroAC50, kroAD50, and kroAE50). Besides the number of colonies studied in the previous sections, extra variants with 10 colonies were also experimented. As already mentioned, this decision was motivated by observation that small number of colonies seem to provide better results. As we will see, this number of colonies also provides good approximations although the best case was obtained with variant ϵD.c with 50 (2×25) colonies.

The statistics (mean and standard deviation) for the cardinality and hyper-volume ratio are collected in Table 22, where, for each of the three problem instances, the non dominated elements of the union of all runs were used as reference sets. From the analysis of the values some conclusions can be made: variant ϵD.c obtained the best hyper-volume for the three problems with over 99% of the hyper-volume of reference. ϵD.c is an ϵ-DANTE variant with 50 colonies (25 for each of the types—see Sects. 4.2, 4.3) and $N_S = 50$ as the limit for the number of solutions built by each ant. Comparing ϵD.c with ϵD.d, where the difference is the imposed limit to the number of solutions, a significant improvement of the hyper-volume can be observed. This improvement is more obvious when comparing the 100 (2×50) colonies variants among themselves (e.g., ϵD.a vs. ϵD.b and ϵD.e vs. ϵD.f). Referring to the cardinality of the approximation set obtained by each of the variants, ϵD.c obtained the larger mean value for kroAC50 and kroAD50, being the second best for the other problem. As already observed in the previous sections, there seems to be a correlation between the cardinality and the hyper-volume ratio/coverage metric, as we will see in the remaining section.

The previous observations are corroborated by the Mann–Whitney test shown in Table 23.[6] According to the tests, ϵD.c was significantly better than the remaining variants. For kroAC50 and kroAD50, variants 2D.b (DANTE) and ϵD.b (ϵ-DANTE) were the second best variants. Those two variants used 20 (2×10) colonies and $N_S = \infty$. For the kroAE50 case, the second best variants were 2D.a (DANTE II) and ϵD.a (ϵ-DANTE), both with 20 colonies and $N_S = 50$.

Table 20 Common parameter values for the complementary experimental setup

	Number of nodes and dimension		
	50 and $m = 2$	50 and $m > 2$	100 and $m = 2$
Ants per cycle	10		
Number of cycles	50		
ρ	0.1		
q_0	0.75		
M	5		
D	30		80
d	40		90
Size of taboo list	10		
Maximum run time	300 s	1,800 s	

Table 21 Parameter values for the complementary study variants

Variant (previously)	Colonies	α	β	ϵ	Version	N_S
D.a (−)	10	1	1	–	DANTE	–
D.b (−)	10	1	5	–	DANTE	–
D.c (D.1)	25	1	1	–	DANTE	–
D.d (D.2)	25	1	5	–	DANTE	–
D.e (D.5)	50	1	1	–	DANTE	–
D.f (D.6)	50	1	5	–	DANTE	–
2D.a (−)	10	5	1	–	DANTE II	50
2D.b (−)	10	5	1	–	DANTE II	∞
2D.c (2D.3)	25	5	1	–	DANTE II	50
2D.d (2D.1)	25	5	1	–	DANTE II	∞
2D.e (2D.4)	50	5	1	–	DANTE II	50
2D.f (2D.30)	50	5	1	–	DANTE II	∞
ϵD.a (−)	10	5	1	0.01	ϵ-DANTE	50
ϵD.b (−)	10	5	1	0.01	ϵ-DANTE	∞
ϵD.c (ϵD.3)	25	5	1	0.01	ϵ-DANTE	50
ϵD.d (ϵD.1)	25	5	1	0.01	ϵ-DANTE	∞
ϵD.e (ϵD.4)	50	5	1	0.01	ϵ-DANTE	50
ϵD.f (ϵD.2)	50	5	1	0.01	ϵ-DANTE	∞

In parentheses is the variant identification previously used, when applicable

[6] For the sake of clarity, only the best eleven variants are presented, which correspond to those with hyper-volume ratio larger than 0.99. We should notice that these variants were the same for every problem instance (kroAC50, kroAD50, and kroAE50).

Table 22 kroAC50, kroAD50, and kroAE50 statistics (mean ans standard deviation) for the cardinality and $\mathcal{S}$ metric

kroAC50									
$\|P\|$									
	D.a	D.b	D.c	D.d	D.e	D.f	2D.a	2D.b	2D.c
Mean	5.050e+02	5.118e+02	5.562e+02	5.142e+02	4.943e+02	4.010e+02	5.258e+02	4.706e+02	4.558e+02
SD	4.869e+01	4.029e+01	1.692e+01	3.416e+01	2.660e+01	3.039e+01	2.880e+01	2.798e+01	2.524e+01
	2D.d	2D.e	2D.f	ϵD.a	ϵD.b	ϵD.c	ϵD.d	ϵD.e	ϵD.f
Mean	2.582e+02	3.200e+02	1.799e+02	5.586e+02	4.998e+02	5.693e+02	3.155e+02	4.788e+02	1.818e+02
SD	4.857e+01	3.480e+01	1.402e+01	2.350e+01	2.437e+01	2.186e+01	3.107e+01	2.561e+01	1.158e+01
$\mathcal{S}$									
	D.a	D.b	D.c	D.d	D.e	D.f	2D.a	2D.b	2D.c
Mean	9.881e-01	9.898e-01	9.935e-01	9.926e-01	9.933e-01	9.914e-01	9.952e-01	9.957e-01	9.949e-01
SD	5.467e-03	3.093e-03	1.346e-03	1.891e-03	1.700e-03	2.463e-03	1.471e-03	1.199e-03	1.118e-03
	2D.d	2D.e	2D.f	ϵD.a	ϵD.b	ϵD.c	ϵD.d	ϵD.e	ϵD.f
Mean	9.822e-01	9.838e-01	9.551e-01	9.948e-01	9.958e-01	9.972e-01	9.895e-01	9.951e-01	9.542e-01
SD	4.642e-03	3.704e-03	3.955e-03	1.242e-03	1.170e-03	6.476e-04	1.964e-03	1.123e-03	3.563e-03
kroAD50									
$\|P\|$									
	D.a	D.b	D.c	D.d	D.e	D.f	2D.a	2D.b	2D.c
Mean	3.903e+02	4.001e+02	3.937e+02	3.758e+02	3.465e+02	3.054e+02	3.825e+02	3.474e+02	3.152e+02
SD	2.125e+01	1.836e+01	1.513e+01	1.762e+01	1.486e+01	2.024e+01	2.290e+01	2.318e+01	1.771e+01
	2D.d	2D.e	2D.f	ϵD.a	ϵD.b	ϵD.c	ϵD.d	ϵD.e	ϵD.f
Mean	2.429e+02	2.483e+02	1.751e+02	4.283e+02	3.706e+02	4.058e+02	2.548e+02	3.284e+02	1.696e+02
SD	2.266e+01	1.598e+01	1.777e+01	2.162e+01	2.258e+01	2.108e+01	1.311e+01	1.896e+01	1.686e+01
$\mathcal{S}$									
	D.a	D.b	D.c	D.d	D.e	D.f	2D.a	2D.b	2D.c
Mean	9.828e-01	9.862e-01	9.890e-01	9.900e-01	9.914e-01	9.899e-01	9.945e-01	9.960e-01	9.935e-01
SD	4.556e-03	4.565e-03	3.925e-03	3.175e-03	1.905e-03	2.024e-03	1.120e-03	1.014e-03	1.417e-03
	2D.d	2D.e	2D.f	ϵD.a	ϵD.b	ϵD.c	ϵD.d	ϵD.e	ϵD.f
Mean	9.854e-01	9.827e-01	9.549e-01	9.950e-01	9.965e-01	9.972e-01	9.878e-01	9.938e-01	9.530e-01
SD	6.057e-03	3.130e-03	3.962e-03	1.411e-03	9.995e-04	1.129e-03	2.263e-03	1.237e-03	4.261e-03
kroAE50									
$\|P\|$									
	D.a	D.b	D.c	D.d	D.e	D.f	2D.a	2D.b	2D.c
Mean	3.850e+02	3.851e+02	4.296e+02	3.846e+02	3.953e+02	3.098e+02	4.408e+02	3.631e+02	3.630e+02
SD	2.632e+01	2.883e+01	2.600e+01	1.924e+01	2.765e+01	2.262e+01	1.809e+01	2.079e+01	2.509e+01
	2D.d	2D.e	2D.f	ϵD.a	ϵD.b	ϵD.c	ϵD.d	ϵD.e	ϵD.f
Mean	2.372e+02	2.719e+02	2.016e+02	4.696e+02	3.896e+02	4.816e+02	2.648e+02	3.932e+02	1.968e+02
SD	1.923e+01	2.096e+01	1.071e+01	1.629e+01	1.973e+01	1.441e+01	1.857e+01	2.281e+01	1.546e+01
$\mathcal{S}$									
	D.a	D.b	D.c	D.d	D.e	D.f	2D.a	2D.b	2D.c
Mean	9.827e-01	9.834e-01	9.928e-01	9.903e-01	9.952e-01	9.921e-01	9.973e-01	9.961e-01	9.957e-01
SD	6.475e-03	5.964e-03	2.501e-03	2.527e-03	1.222e-03	2.389e-03	8.958e-04	8.922e-04	1.206e-03
	2D.d	2D.e	2D.f	ϵD.a	ϵD.b	ϵD.c	ϵD.d	ϵD.e	ϵD.f
Mean	9.814e-01	9.854e-01	9.591e-01	9.978e-01	9.970e-01	9.987e-01	9.888e-01	9.968e-01	9.585e-01
SD	2.999e-03	3.478e-03	3.726e-03	6.764e-04	6.112e-04	2.650e-04	4.313e-03	6.617e-04	3.070e-03

Comparing the ϵ-DANTE variants with the remainder (the horizontal and vertical lines in the table do the separation), in the majority of the cases the Mann–Whitney test return p values smaller than 0.05, which allow to reject the null hypothesis, accepting that the median of the hypervolume ratio should be larger for ϵ-DANTE variants than

Table 23 Mann–Whitney test results for the 11 best variants (the ones with hyper-volume ratio larger than 0.99) over kroAC50, kroAD50, and kroAE50 problems

kroAC50										
D.c	▲ 0.08	▲ 0.33	▽ 0.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00
	(− 0.0;∞)	(− 0.0;∞)	(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▲ 0.93	D.d	▲ 0.80	▽ 0.04	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00
(− 0.0;∞)		(− 0.0;∞)	(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▲ 0.68	▲ 0.21	D.e	▽ 0.01	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00
(−0.0; ∞)	(− 0.0;∞)		(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▲ 1.00	▲ 0.96	▲ 0.99	D.f	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00
(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)		(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	2D.a	▲ 0.88	▲ 0.22	▲ 0.14	▲ 0.93	▲ 1.00	▲ 0.39
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)		(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▲ 0.12	2D.b	▽ 0.01	▽ 0.01	▲ 0.70	▲ 1.00	▽ 0.04
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(− 0.0;∞)		(0.0;∞)	(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▲ 0.79	▲ 0.99	2D.c	▲ 0.42	▲ 1.00	▲ 1.00	▲ 0.74
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)		(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▲ 0.86	▲ 0.99	▲ 0.58	ϵ D.a	▲ 1.00	▲ 1.00	▲ 0.80
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)		(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▲ 0.07	▲ 0.30	▽ 0.00	▽ 0.00	ϵ D.b	▲ 1.00	▽ 0.01
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(0.0;∞)	(0.0;∞)		(− 0.0;∞)	(0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	ϵD.c	▽ 0.00
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)		(0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▲ 0.62	▲ 0.96	▲ 0.26	▲ 0.20	▲ 0.99	▲ 1.00	ϵ D.e
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	
kroAD50										
D.c	▲ 0.76	▲ 0.99	▲ 0.72	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00
	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▲ 0.24	D.d	▲ 0.95	▲ 0.32	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00
(−0.0; ∞)		(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▽ 0.01	▲ 0.05	D.e	▽ 0.01	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00
(0.0;∞)	(− 0.0;∞)		(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▲ 0.28	▲ 0.69	▲ 0.99	D.f	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00
(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)		(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	2D.a	▲ 1.00	▽ 0.00	▲ 0.92	▲ 1.00	▲ 1.00	▽ 0.04
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)		(− 0.0;∞)	(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	2D.b	▽ 0.00	▽ 0.00	▲ 0.97	▲ 1.00	▽ 0.00
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)		(0.0;∞)	(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▲ 1.00	▲ 1.00	2D.c	▲ 1.00	▲ 1.00	▲ 1.00	▲ 0.84
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)		(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▲ 0.08	▲ 1.00	▽ 0.00	ϵ D.a	▲ 1.00	▲ 1.00	▽ 0.00
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(0.0;∞)		(− 0.0;∞)	(− 0.0;∞)	(0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.04	▽ 0.00	▽ 0.00	ϵD.b	▲ 0.99	▽ 0.00
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)		(− 0.0;∞)	(0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.01	ϵ D.c	▽ 0.00
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)		(0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▲ 0.96	▲ 1.00	▲ 0.17	▲ 1.00	▲ 1.00	▲ 1.00	ϵD.e
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	
kroAE50										
D.c	▽ 0.00	▲ 1.00	▲ 0.19	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00
	(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)

Table 23 continued

▲ 1.00	D.d	▲ 1.00	▲ 0.99	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00
(− 0.0;∞)		(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▽ 0.00	▽ 0.00	D.e	▽ 0.00	▲ 1.00	▲ 1.00	▲ 0.94	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00
(0.0;∞)	(0.0;∞)		(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▲ 0.82	▽ 0.01	▲ 1.00	D.f	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00
(− 0.0;∞)	(0.0;∞)	(− 0.0;∞)		(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	2D.a	▽ 0.00	▽ 0.00	▲ 0.94	▽ 0.01	▲ 1.00	▽ 0.00
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)		(0.0;∞)	(0.0;∞)	(− 0.0;∞)	(0.0;∞)	(− 0.0;∞)	(0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▲ 1.00	2D.b	▲ 0.11	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(− 0.0;∞)		(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▽ 0.00	▽ 0.00	▲ 0.06	▽ 0.00	▲ 1.00	▲ 0.90	2D.c	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00
(0.0;∞)	(0.0;∞)	(− 0.0;∞)	(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)		(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▲ 0.06	▽ 0.00	▽ 0.00	ϵ D.a	▽ 0.00	▲ 1.00	▽ 0.00
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(− 0.0;∞)	(0.0;∞)	(0.0;∞)		(0.0;∞)	(− 0.0;∞)	(0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▲ 0.99	▽ 0.00	▽ 0.00	▲ 1.00	ϵD.b	▲ 1.00	▲ 0.28
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(− 0.0;∞)	(0.0;∞)	(0.0;∞)	(− 0.0;∞)		(− 0.0;∞)	(− 0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	ϵD.c	▽ 0.00
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)		(0.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▲ 1.00	▽ 0.00	▽ 0.00	▲ 1.00	▲ 0.73	▲ 1.00	ϵD.e
(0.0;∞)	(0.0;∞)	(0.0;∞)	(0.0;∞)	(− 0.0;∞)	(0.0;∞)	(0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	(− 0.0;∞)	

The horizontal/vertical lines delimit the ϵ-DANTE variants from the others. For each pair of variants the Mann–Whitney test p value, the confidence interval, and the symbology ▽|▲ indicating that we should reject the null hypothesis, or the data does not allow us to conclude that the medians are different, respectively, are presented

for the others. The major exception is ϵD.e (100 colonies) which was not in general significantly better than the DANTE and DANTE II variants.

Referring to the limit for the number of solutions that an ant can build, N_S, 7 of the 11 selected variants used $N_S = 50$. Although the best variant was ϵD.c with 50 (2×25) colonies, the second best variant used the smallest number of colonies (10 and 20 for the DANTE and DANTE II cases, respectively).

The beanplots with the distribution of the coverage metric (C) for the three instances are sketched in Fig. 10. For example, the rectangle in the intersection of row 1 with column 2 presents three beanplots (one for each of the kroAC50, kroAD50, and kroAE50 instances) for the C distribution of D.a against D.b, that is, the percentage of elements of D.b which are dominated by elements of D.a. The intersection of row 2 with column 1 presents the reverse distribution of percentages, that is, the percentages of elements of D.b which are dominated by elements of D.a. The observation of these beanplots completes the analysis, and emphasizes the previous observations, that is, ϵD.a, ϵD.b, and ϵD.c (both ϵ-DANTE) are among the best variants. The worst variants were 2D.f (DANTE) and ϵD.f (ϵ-DANTE), both with 50 colonies and $N_S = \infty$. Compared with the remaining cases, 2D.f and ϵD.f obtained a very small cardinality, which is associated with the small number of cycles completed in the available time. This small number of cycles does not allow a proper exchange of information between the colonies leading to a smaller exploitation of the search space.

5.2 Problems with more than 50 nodes or more than 2 objectives

The results for the kroABC50 (three objectives), kroABCD50 (four objectives), and kroAB100 are presented in this section. To simplify the analysis, the presented results are for three variants: V_{ACO},V_{DANTE} and $V_{\epsilon-\text{DANTE}}$. The common parameter values were already presented in Table 20. The remaining parameter values were set according with the best variants of Version 2, DANTE, and ϵ-DANTE, that is:

V_{ACO} and V_{DANTE} : 50 colonies and $\alpha_{\max} = \beta_{\max} = 1$.
$V_{\epsilon-\text{DANTE}}$: 25 colonies, $\alpha_{\max} = 5$,$\beta_{\max} = 1$,N_S equal to the number of nodes of the network (50 or 100), and $\epsilon = 0.01$.

Table 24 presents the C coverage metric statistics (mean and standard deviation), comparing the three variants over the three problems in analysis. The $V_{\epsilon-\text{DANTE}}$ was the best variant for all problems. For example, for kroAB100 more than 97% of the solutions obtained by V_{ACO} and V_{DANTE} are weakly dominated by solutions of $V_{\epsilon-\text{DANTE}}$. Although not so evident, the same superiority is maintained for

Fig. 10 *C* metric values beanplots for kroAC50, kroAD50, and kroAE50. The *rectangle* in the intersection of row *i* with column *j* presents three beanplots (one for each of the kroAC50, kroAD50, and kroAE50 instances) for the *C* distribution of variant *i* against variant *j*, that is, the percentage of elements returned by variant *j* which are weakly dominated by elements returned by variant *i*

kroABC50 and kroABCD50, where at most 32.1% of the solutions obtained by $V_{\epsilon-\mathrm{DANTE}}$ are weakly dominated by solutions of the other two variants.

This study would not be complete without the comparison between the proposed method and other state-of-the-art algorithms, which will be made in the next section.

Table 24 C metric statistics (mean and standard deviation) for kroAB100, kroABC50, kroABCD50

kroAB100		
V_{ACO}	0.164 (0.12)	0.008 (0.01)
0.699 (0.19)	V_{DANTE}	0.016 (0.03)
0.977 (0.04)	0.973 (0.06)	$V_{\epsilon-\text{DANTE}}$
kroABC50		
V_{ACO}	0.628 (0.12)	0.314 (0.08)
0.283 (0.11)	V_{DANTE}	0.275 (0.08)
0.576 (0.06)	0.660 (0.06)	$V_{\epsilon-\text{DANTE}}$
kroABCD50		
V_{ACO}	0.888 (0.03)	0.321 (0.06)
0.055 (0.03)	V_{DANTE}	0.144 (0.05)
0.609 (0.05)	0.833 (0.04)	$V_{\epsilon-\text{DANTE}}$

6 Comparing with other heuristics

In the previous sections, it was concluded that, for the studied variants, 2M.7 (Sect. 3.3) was the best performing variant not using an explicit local search, while ϵD.3[7] (Sect. 4.3) was the best among the ones using the DANTE's oriented depth search. However, no conclusive deduction could have been made about the quality of the proposed methods when compared with other heuristics, and in particular with other MOACO algorithms. Therefore, this section will be dedicated to the comparison of those two variants with three known heuristics, namely:

- Multiple Ant Colony System (MACS) (Barán and Schaerer 2003). MACS is an adaptation of the Multiple Ant Colony System for Vehicle Routing Problem with Time Windows (MACS-VRPTW) (Gambardella et al. 1999). In (García-Martínez et al. 2007), MACS was compared against eight other MOACO algorithms and the authors selected it as the one with the best overall performance covering a set of MOTSP instances.
- Non-dominated Sorting Genetic Algorithm II (NSGA-II) (Deb et al. 2000). NSGA-II is one of the most popular multiple objective evolutionary algorithms that uses an elitist approach (Deb 2001). Its fitness assignment scheme consists in sorting the population in different fronts using the non-domination order relation. Then, to form the next generation, the algorithm combines the current population and its offspring generated with the standard bimodal crossover and polynomial operators. Finally, the best individuals in terms of non-dominance and diversity are chosen. In our case, the selection of NSGA-II follows the same line of reasoning used in (García-Martínez et al. 2007).
- Memetic-Pareto Archived Evolution Strategy (M-PAES) (Knowles and Corne 2000b). M-PAES is based on (1 + 1)-PAES which is a local search MOO algorithm (Knowles and Corne 2000a), but uses a population of solutions and periodically employs crossover to recombine the distinct local optima found using the (1 + 1)-PAES procedure. Algorithm 5 describes our adaptation of the (1 + 1)-PAES to the MOTSP problem, where the mutation of the solutions is achieved through the swapping of two randomly chosen elements of the cycle. The selection of MPAES algorithm is based in the fact that, as we will see below, the (1 + 1)-PAES was the local search procedure used to hybridize MACS and 2M.7. Therefore, it seems interesting to have a comparison of the studied methods with one of the state-of-the-art methods that uses (1 + 1)-PAES.

As already suggested, the (1 + 1)-PAES local search procedure was used to obtain two hybrid methods: MACS + PAES and 2M.7 + PAES. In both cases, (1 + 1)-PAES (Algorithm 5) was applied to each solution generated by MACS and 2M.7, respectively. This will allows us to compare the ϵ-DANTE with other two MOACO algorithms complemented with a local search.

Algorithm 5 Adaptation of the (1+1)-PAES to the MOTSP problem.

Require: $\mathcal{P}$, C ▷ Archive and an approximation cycle
$NF = 0$ ▷ NF – number of fails
$NM = 0$ ▷ NM – number of moves
while NF is smaller than the maximum number of fails and the maximum number of moves was not reached **do**
  $NM = NM + 1$
  Swap two elements in cycle C to produce M ▷ Mutate C
  Evaluate M
  if C dominates M **then**
    Discard M
    $NF = NF + 1$
  else
    if M dominates C **then**
      Replace C with M, add M to $\mathcal{P}$, and reset $NF = 0$
    else
      if M is dominated by any member of $\mathcal{P}$ **then**
        Discard M
      else
        Apply $test(C,M,\mathcal{P})$ to determine which becomes the new current solution and whether to add M to the archive ▷ $test$ is a diversity function defined as in (Knowles and Corne, 2000b).
      end if
    end if
  end if
end while

To compare the performance of the the seven methods (MACS, MACS + PAES, NSGA-II, MPAES, 2M.7,

[7] Although ϵD.1 was considered the best variant in Sect. 4.3, ϵD.3 $\equiv\epsilon$Dc was the best performing variant for a broader set of instances (see Sect. 5).

Table 25 Parameter values for the MACS, PAES, NSGA-II, and MPAES methods

	Value	Observations
Problems	kroAB50	
	kroAC50	
	kroAD50	
	kroAB100	
Anti-reference solution	(9*e*4, 9*e*4)	50 nodes instances
	(1.8*e*5, 1.8*e*5)	100 nodes instances
Maximum run time	300 s	50 nodes instances
	1,800 s	100 nodes instances
Number of runs	25	
Ants per cycle	20	MACS, MACS + PAES
α	1	
β	2	
ρ	0.2	
$q0$	0.98	
Population size	100	NSGA-II, MPAES, (1 + 1)-PAES
Crossover probability	0.8	
Mutation probability	0.1	
(1 + 1)-PAES maximum number of: fails and moves	50	50 nodes instances
	100	100 nodes instances

2M.7 + PAES, ϵD.3), four instances of the MOTSP (kroAB50, kroAC50, kroAD50, and kroAB100) were considered. The parameter values for 2M.7 and ϵD.3 were presented in Tables 7 and 16, respectively. The remaining parameter values are presented in Table 25. As in the previous sections, all methods were implemented in C++, compiled using gcc 4.3.3, and run on Ubuntu 7.10 over a 3.0 GHz computer with 1 GB of RAM.

Table 26 summarizes the statistics for the cardinality and $\mathcal{S}$ metrics, and Table 27 complements these statistics by presenting the Mann–Whitney test results. From the analysis of those tables some conclusions can be made:

- Overall tested instances, the best mean hyper-volume was obtained by ϵ-DANTE.
- Except for kroAB100 instance, NSGA-II and MPAES were the second best methods referring to the hyper-volume metric. For the kroAB100 instance, MACS and MACS + PAES were better than NSGA-II and MPAES.
- Except for kroAC50 instance, the largest cardinality was obtained by MPAES.
- Except for kroAB50 instance, 2M.7 was the method with smallest mean hyper-volume and was among the worst methods referring to cardinality.
- In general, the use of the (1 + 1)-PAES local search, both in MACS and in 2M.7, improved the results obtained without the local search. The biggest exception was 2M.7 + PAES for kroAB50 instance, were the results obtained with 2M.7 + PAES were significantly worst.
- The Mann–Whitney test results corroborate the fact that ϵ-DANTE was the best method when it refers to the hyper-volume metric. In general, for the 50 nodes instances, the same test shows that MPAES and NSGA-II where the second best methods. For the 100 nodes instance, MACS + PAES was the second best method. Except for kroAB50, 2M.7 was statistically worst than all the other methods.

To complete our study, Table 28 shows the C coverage metric statistics (mean and standard deviation). The corresponding beanplots, with the distribution of the coverage metric, are sketched in Fig. 11. From the analysis of the coverage values, some conclusions can be made (over/for the studied instances):

- The ϵ-DANTE method, ϵD.3, obtains the best coverage values overall instances. The worst value is obtained when compared with MPAES in instance kroAB50, where a mean of 25% of the solutions of ϵD.3 are weakly dominated by solutions of MPAES, against the 66% of solutions of MPAES weakly dominated by the ones of ϵD.3.
- NSGA-II an MPAES are in general better than MACS, MACS + PAES, 2M.7, and 2M.7 + PAES.
- The comparison of 2M.7 with 2M.7 + PAES is not conclusive and problem dependent. Furthermore, from the analysis of the beanplots, it is observable that the percentage of weakly dominated elements returned by 2M.7 + PAES has a large spread. A possible reason for the non improvement of the results, might be an excessive intensity of the local search, which will consume resources necessary to do the exploration of the search space.
- The use of the local search in MACS + PAES has improved the results obtained with MACS.
- Comparing 2M.7 against MACS and MACS + PAES, we can observe that, except for kroAB50, the latest methods have better coverage results. Comparing 2M.7 + PAES against MACS and MACS + PAES does not show an improvement of the results (they are even worsen for kroAB50).

Figures 12 and 13 sketch typical objective fronts for each of the methods over the four instances in analysis.

Table 26 Statistics (mean ans standard deviation) for the cardinality and $\mathcal{S}$ metric of the seven methods (MACS, MACS + PAES, NSGA-II, MPAES, 2M.7, 2M.7 + PAES, ϵD.3) over four instances of the MOTSP (kroAB50, kroAC50, kroAD50, and kroAB100)

	MACS	MACS + PAES	NSGA-II	MPAES	2M.7	2M.7 + PAES	ϵD.3
kroAB50							
$\|P\|$							
Mean	9.644e+01	2.022e+02	1.230e+02	**6.324e+02**	2.227e+02	8.136e+01	3.143e+02
SD	8.761e+00	1.550e+01	9.855e+00	9.130e+01	2.781e+01	1.030e+01	3.108e+01
$\mathcal{S}$							
Mean	9.513e−01	9.663e−01	9.737e−01	9.700e−01	9.804e−01	8.711e−01	**9.931e−01**
SD	4.590e−03	4.775e−03	4.254e−03	1.633e−02	2.794e−03	1.443e−02	1.409e−03
kroAC50							
$\|P\|$							
Mean	1.082e+02	2.263e+02	1.043e+02	5.201e+02	7.236e+01	7.424e+01	**5.696e+02**
SD	1.064e+01	2.877e+01	1.019e+01	6.893e+01	7.222e+00	8.368e+00	2.163e+01
$\mathcal{S}$							
Mean	9.408e−01	9.552e−01	9.696e−01	9.704e−01	8.644e−01	8.627e−01	**9.976e−01**
SD	3.545e−03	3.654e−03	5.705e−03	1.318e−02	1.635e−02	1.108e−02	6.363e−04
kroAD50							
$\|P\|$							
Mean	9.712e+01	2.081e+02	1.086e+02	**4.407e+02**	6.984e+01	8.604e+01	4.061e+02
SD	6.616e+00	1.648e+01	9.019e+00	3.574e+01	9.797e+00	1.196e+01	2.122e+01
$\mathcal{S}$							
Mean	9.364e−01	9.538e−01	9.694e−01	9.741e−01	8.378e−01	8.480e−01	**9.978e−01**
SD	4.233e−03	3.077e−03	4.089e−03	1.095e−02	1.307e−02	1.104e−02	8.382e−04
kroAB100							
$\|P\|$							
Mean	1.147e+02	2.884e+02	7.256e+01	**1.168e+03**	7.724e+01	9.292e+01	1.107e+03
SD	7.481e+00	2.122e+01	8.322e+00	1.516e+02	9.148e+00	1.629e+01	8.669e+01
$\mathcal{S}$							
Mean	9.228e−01	9.362e−01	9.089e−01	9.244e−01	7.505e−01	7.881e−01	**9.945e−01**
SD	3.366e−03	2.464e−03	4.147e−03	1.188e−02	1.496e−02	1.851e−02	1.194e−03

Best mean value for each case is shown in bold type

7 Conclusions and future work

This paper presents a hybrid meta-heuristic called ϵ-Depth ANT Explorer (ϵ-DANTE) that uses an efficient and different local search technique, adapted to a pheromone-oriented procedure. In each one of the ϵ-DANTE cycles, sets of solutions are computed using a first mandatory and a second elective phases:

- In the first phase a solution is generated based on a constructive procedure that shapes each solution by successively adding selected components. This procedure uses (multiple) pheromone trails and heuristics to enhance the quality of the solutions.
- The second phase is triggered when a solution is within an ϵ range of the approximation set (the collection of the best performing solutions), or is not dominated by any solution in this set. The same (multiple) pheromone trails and heuristics used in the first phase, are used here to guide a limited/oriented depth search method, including the more promising components into the solutions under construction.

It is common to use greedy pheromone update strategies in ACO algorithms, that is, to adopt procedures where only the best performing solutions are used to update the pheromone trails. For the multiple objective case, one possible procedure uses all the elements of the approximation set in this update. However, the cardinality of this set is usually very large, which introduces "noisy" pheromone trails and consequently reduces the performance of the algorithm. Therefore, a complementary pheromone update strategy, that uses as a base the elements of the approximation set, has also been proposed. The proposed update strategy uses only smaller subsets of the approximation set, based on its selective partition, which successively favors the exploration of small parts of the search space.

Table 27 Mann–Whitney test results for the seven methods (MACS, MACS + PAES, NSGA-II, MPAES, 2M.7, 2M.7 + PAES, ϵD.3) over four instances of the MOTSP (kroAB50, kroAC50, kroAD50, and kroAB100)

kroAB50						
MACS	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▽ 0.00	▲ 1.00
	(.0;∞)	(.0;∞)	(.0;∞)	(.0;∞)	(0.1;∞)	(.0;∞)
▽ 0.00	MACS	▲ 1.00	▲ 0.97	▲ 1.00	▽ 0.00	▲ 1.00
(.0;∞)	+PAES	(.0;∞)	(.0;∞)	(.0;∞)	(0.1;∞)	(.0;∞)
▽ 0.00	▽ 0.00	NSGA-II	▲ 0.18	▲ 1.00	▽ 0.00	▲ 1.00
(.0;∞)	(.0;∞)		(.0;∞)	(.0;∞)	(0.1;∞)	(.0;∞)
▽ 0.00	▽ 0.04	▲ 0.82	MPAES	▲ 1.00	▽ 0.00	▲ 1.00
(.0;∞)	(.0;∞)	(.0;∞)		(.0;∞)	(0.1;∞)	(.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	2M.7	▽ 0.00	▲ 1.00
(.0;∞)	(.0;∞)	(.0;∞)	(.0;∞)		(0.1;∞)	(.0;∞)
▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	2M.7+	▲ 1.00
(− .1;∞)	(− .1;∞)	(− .1;∞)	(− .1;∞)	(− .1;∞)	PAES	(− .1;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	ϵ D.3
(.0;∞)	(.0;∞)	(.0;∞)	(.0;∞)	(.0;∞)	(0.1;∞)	
kroAC50						
MACS	▲ 1.00	▲ 1.00	▲ 1.00	▽ 0.00	▽ 0.00	▲ 1.00
	(.0;∞)	(.0;∞)	(.0;∞)	(0.1;∞)	(0.1;∞)	(− .1;∞)
▽ 0.00	MACS	▲ 1.00	▲ 1.00	▽ 0.00	▽ 0.00	▲ 1.00
(.0;∞)	+PAES	(.0;∞)	(.0;∞)	(0.1;∞)	(0.1;∞)	(.0;∞)
▽ 0.00	▽ 0.00	NSGA-II	▲ 0.67	▽ 0.00	▽ 0.00	▲ 1.00
(.0;∞)	(.0;∞)		(.0;∞)	(0.1;∞)	(0.1;∞)	(.0;∞)
▽ 0.00	▽ 0.00	▲ 0.34	MPAES	▽ 0.00	▽ 0.00	▲ 1.00
(.0;∞)	(.0;∞)	(.0;∞)		(0.1;∞)	(0.1;∞)	(.0;∞)
▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	2M.7	▲ 0.26	▲ 1.00
(− .1;∞)	(− .1;∞)	(− .1;∞)	(− .1;∞)		(.0;∞)	(− .1;∞)
▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▲ 0.74	2M.7+	▲ 1.00
(− .1;∞)	(− .1;∞)	(− .1;∞)	(− .1;∞)	(.0;∞)	PAES	(− .1;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	ϵD.3
(0.1;∞)	(.0;∞)	(.0;∞)	(.0;∞)	(0.1;∞)	(0.1;∞)	
kroAD50						
MACS	▲ 1.00	▲ 1.00	▲ 1.00	▽ 0.00	▽ 0.00	▲ 1.00
	(.0;∞)	(.0;∞)	(.0;∞)	(0.1;∞)	(0.1;∞)	(− .1;∞)
▽ 0.00	MACS	▲ 1.00	▲ 1.00	▽ 0.00	▽ 0.00	▲ 1.00
(.0;∞)	+PAES	(.0;∞)	(.0;∞)	(0.1;∞)	(0.1;∞)	(.0;∞)
▽ 0.00	▽ 0.00	NSGA-II	▲ 1.00	▽ 0.00	▽ 0.00	▲ 1.00
(.0;∞)	(.0;∞)		(.0;∞)	(0.1;∞)	(0.1;∞)	(.0;∞)
▽ 0.00	▽ 0.00	▽ 0.00	MPAES	▽ 0.00	▽ 0.00	▲ 1.00
(.0;∞)	(.0;∞)	(.0;∞)		(0.1;∞)	(0.1;∞)	(.0;∞)
▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	2M.7	▲ 0.99	▲ 1.00
(− .1;∞)	(− .1;∞)	(− .1;∞)	(− .1;∞)		(.0;∞)	(− .2;∞)
▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▽ 0.01	2M.7+	▲ 1.00
(−.1; ∞)	(− .1;∞)	(− .1;∞)	(− .1;∞)	(.0;∞)	PAES	(− .2;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	ϵD.3
(0.1;∞)	(.0;∞)	(.0;∞)	(.0;∞)	(0.2;∞)	(0.1;∞)	
kroAB100						
MACS	▲ 1.00	▽ 0.00	▲ 0.62	▽ 0.00	▽ 0.00	▲ 1.00
	(.0;∞)	(.0;∞)	(.0;∞)	(0.2;∞)	(0.1;∞)	(− .1;∞)
▽ 0.00	MACS	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▲ 1.00
(.0;∞)	+PAES	(.0;∞)	(.0;∞)	(0.2;∞)	(0.1;∞)	(− .1;∞)
▲ 1.00	▲ 1.00	NSGA-II	▲ 1.00	▽ 0.00	▽ 0.00	▲ 1.00
(.0;∞)	(.0;∞)		(.0;∞)	(0.2;∞)	(0.1;∞)	(.0;∞)
▲ 0.39	▲ 1.00	▽ 0.00	MPAES	▽ 0.00	▽ 0.00	▲ 1.00
(.0;∞)	(.0;∞)	(.0;∞)		(0.2;∞)	(0.1;∞)	(− .1;∞)
▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	2M.7	▲ 1.00	▲ 1.00
(− .2;∞)	(− .2;∞)	(− .2;∞)	(− .2;∞)		(.0;∞)	(− .3;∞)
▲ 1.00	▲ 1.00	▲ 1.00	▲ 1.00	▽ 0.00	2M.7+	▲ 1.00
(− .1;∞)	(− .2;∞)	(− .1;∞)	(− .1;∞)	(.0;∞)	PAES	(− .2;∞)
▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	▽ 0.00	ϵD.3
(0.1;∞)	(0.1;∞)	(0.1;∞)	(0.1;∞)	(0.2;∞)	(0.2;∞)	

For each pair of variants the Mann–Whitney test p value, the confidence interval, and the symbology ▽|▲ indicating that we should reject the null hypothesis, or the data does not allow us to conclude that the medians are different, respectively, are presented

Versions of DANTE (I and II), also proposed in this contribution, and ϵ-DANTE were implemented and applied to a set of instances of the MOTSP. We verified that the methods converge towards the reference front, obtaining good results. Furthermore, both methods significantly improve the results obtained with the MOACO. This fact was corroborated by the use of reference metrics: coverage, hyper-volume, $R1$, and $R3$ (Fig. 14).

Variants of our best performing MOACO without local search and ϵ-DANTE (2M.7 and ϵD.3, respectively) were then compared with other heuristics (MACS, NSGA-II, and MPAES) over four instances of the MOTSP. It was concluded that, for that set of instances, ϵ-DANTE was the best performing method overall. Furthermore, ϵ-DANTE was compared with other two MOACO with local search (MACS + PAES and 2M.7 + PAES). Again, under the same conditions, ϵ-DANTE was much better then the other MOACO with local search.

The main idea of this contribution was (1) to do the analysis of the reliability, accuracy and versatility of the DANTE's and ϵ-DANTE's methods, and (2) to prove that both methods can be used as a high performance tool in optimization problems, namely in problems for which their intrinsic complexity is recognized.

In general, from the analysis of the developed work, a set of future researches can derive. Although the promising results obtained by DANTE and ϵ-DANTE, its effective application is significantly limited by the size of the problems. Therefore, the inclusion in the search process of other optimization strategies (e.g., the parallelization of the process) and heuristics, should be thought as a way to solve larger instances, to further improve the quality of the solutions, and to accelerate the search procedure. Likewise, the pheromone-updating step is fundamental in the search procedure, which implies that further improvements and possible alternatives should be thought. Another line of development is the application of these methods to other

Table 28 *C* metric statistics (mean and standard deviation) comparing seven methods (MACS, MACS + PAES, NSGA-II, MPAES, 2M.7, 2M.7 + PAES, ϵD.3) over four instances of the MOTSP (kroAB50, kroAC50, kroAD50, and kroAB100)

kroAB50						
MACS	0.14 (0.09)	0.07 (0.05)	0.10 (0.12)	0.01 (0.02)	0.49 (0.23)	0.00 (0.00)
0.80 (0.11)	MACS + PAES	0.20 (0.07)	0.24 (0.18)	0.04 (0.04)	0.67 (0.19)	0.01 (0.02)
0.80 (0.10)	0.59 (0.12)	NSGA-II	0.42 (0.26)	0.26 (0.08)	0.90 (0.13)	0.15 (0.06)
0.82 (0.17)	0.68 (0.20)	0.47 (0.24)	MPAES	0.36 (0.23)	0.90 (0.17)	0.25 (0.19)
0.97 (0.04)	0.89 (0.05)	0.47 (0.09)	0.51 (0.23)	2M.7	0.96 (0.08)	0.21 (0.13)
0.20 (0.10)	0.13 (0.09)	0.04 (0.05)	0.06 (0.10)	0.01 (0.02)	2M.7 + PAES	0.00 (0.00)
1.00 (0.00)	0.98 (0.03)	0.65 (0.11)	0.66 (0.22)	0.67 (0.16)	0.99 (0.01)	ϵD.3
kroAC50						
MACS	0.14 (0.08)	0.11 (0.06)	0.14 (0.10)	0.24 (0.17)	0.39 (0.17)	0.00 (0.00)
0.79 (0.08)	MACS + PAES	0.21 (0.07)	0.26 (0.10)	0.38 (0.21)	0.54 (0.18)	0.02 (0.03)
0.70 (0.11)	0.52 (0.11)	NSGA-II	0.34 (0.24)	0.96 (0.05)	0.99 (0.02)	0.01 (0.02)
0.77 (0.12)	0.61 (0.16)	0.59 (0.25)	MPAES	0.97 (0.05)	0.99 (0.02)	0.09 (0.12)
0.30 (0.09)	0.24 (0.09)	0.02 (0.03)	0.01 (0.04)	2M.7	0.64 (0.21)	0.00 (0.00)
0.23 (0.08)	0.17 (0.07)	0.00 (0.01)	0.00 (0.02)	0.26 (0.19)	2M.7 + PAES	0.00 (0.00)
1.00 (0.01)	0.99 (0.02)	0.96 (0.05)	0.93 (0.07)	1.00 (0.00)	1.00 (0.00)	ϵD.3
kroAD50						
MACS	0.10 (0.06)	0.09 (0.05)	0.09 (0.07)	0.27 (0.07)	0.32 (0.14)	0.00 (0.00)
0.84 (0.06)	MACS + PAES	0.23 (0.06)	0.18 (0.12)	0.48 (0.25)	0.53 (0.17)	0.00 (0.00)
0.78 (0.05)	0.58 (0.06)	NSGA-II	0.30 (0.22)	1.00 (0.00)	1.00 (0.00)	0.02 (0.02)
0.84 (0.07)	0.72 (0.12)	0.60 (0.22)	MPAES	1.00 (0.01)	1.00 (0.00)	0.08 (0.08)
0.26 (0.15)	0.17 (0.25)	−0.00 (0.00)	0.00 (0.01)	2M.7	0.46 (0.30)	0.00 (0.00)
0.27 (0.14)	0.18 (0.17)	−0.00 (0.00)	−0.00 (0.00)	0.47 (0.30)	2M.7 + PAES	0.00 (0.00)
1.00 (0.00)	0.99 (0.00)	0.95 (0.02)	0.91 (0.08)	1.00 (0.00)	1.00 (0.00)	ϵD.3
kroAB100						
MACS	0.08 (0.06)	0.37 (0.06)	0.20 (0.09)	0.88 (0.18)	0.37 (0.20)	0.00 (0.00)
0.88 (0.07)	MACS + PAES	0.53 (0.10)	0.27 (0.09)	0.98 (0.06)	0.75 (0.24)	0.00 (0.00)
0.46 (0.07)	0.33 (0.08)	NSGA-II	0.09 (0.07)	0.98 (0.06)	0.87 (0.16)	0.00 (0.00)
0.63 (0.11)	0.59 (0.09)	0.86 (0.09)	MPAES	1.00 (0.00)	0.99 (0.03)	0.00 (0.00)
0.04 (0.05)	0.00 (0.01)	0.00 (0.02)	0.00 (0.00)	2M.7	0.06 (0.09)	0.00 (0.00)
0.21 (0.08)	0.08 (0.08)	0.06 (0.08)	0.01 (0.02)	0.85 (0.14)	2M.7 + PAES	0.00 (0.00)
1.00 (0.00)	0.99 (0.02)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	ϵD.3

practical problems, namely problems where ACO algorithms have proven difficulties to successfully exploit the search space. Finally, we would like to do a more profound analysis of the influence of the ϵ parameter value in the performance of the ϵ-DANTE algorithm, as well as present alternative approaches (e.g., dynamically adapt ϵ to the problems instances and optimization phases).

Acknowledgments We thank the Engineering Institute of the University of Algarve, in particular to the Department of Electrical Engineering, the Department of Civil Engineering, and the Centro de Simulação e Cálculo (CsC) for the resources, the projects Optimización de redes de interconexión (MTM2008-05866-C03-01) and Matemática Discreta en Andalucia (MaDiscA) (P06-FQM-01649), and the help for the consolidation of the research group FQM-164 (2008/FQM-164). A special thanks to the reviewers who significantly helped to improve this contribution and to the authors of García-Martínez et al. (2007) for providing us the base codes for some of the methods presented in Sect. 6.

Appendix A: Performance metrics

In (Zitzler et al. 2002, 2003) it is shown that there is no combination of unary quality measures that, in general, can indicate that an approximation $\mathcal{P}$ is better than an approximation $\mathcal{Q}$. Nevertheless, these quality indicators are widely used namely as combinations of several of them. These combinations should be based on the distinct characteristics of each one of those operators, and should measure the quality of the methods in some desirable aspects (Bui et al. 2001; Deb 2001):

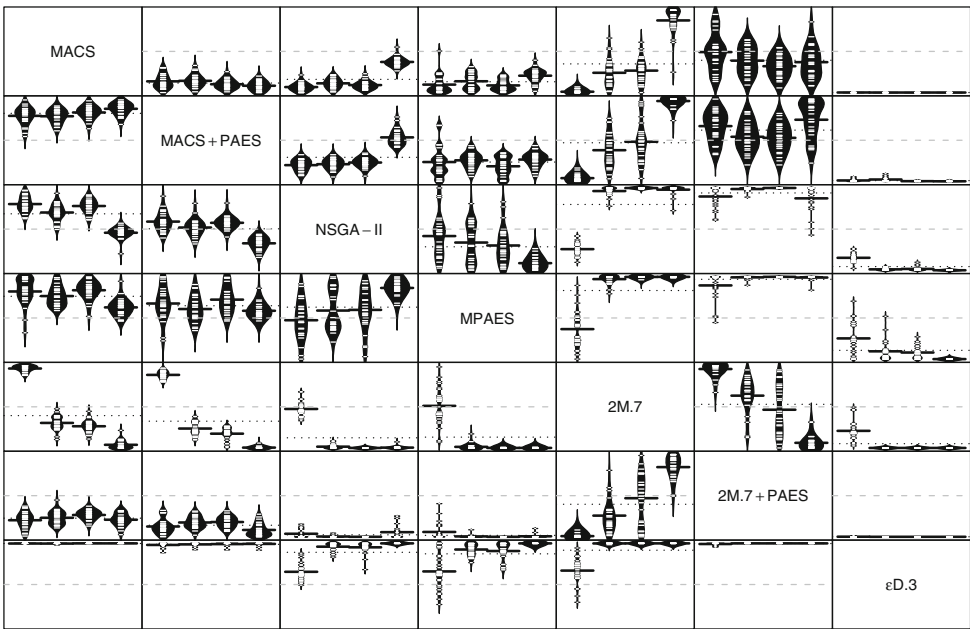

Fig. 11 *C* metric values beanplots comparing seven methods (MACS, MACS + PAES, NSGA-II, MPAES, 2M.7, 2M.7 + PAES, ϵD.3) over four instances of the MOTSP (kroAB50, kroAC50, kroAD50, and kroAB100). The *rectangle* in the intersection of row *i* with column *j* presents four beanplots (one for each of the kroAB50, kroAC50, kroAD50, and kroAB100 instances) for the *C* distribution of variant *i* against variant *j*, that is, the percentage of elements returned by variant *j* which are weakly dominated by elements returned by variant *i*

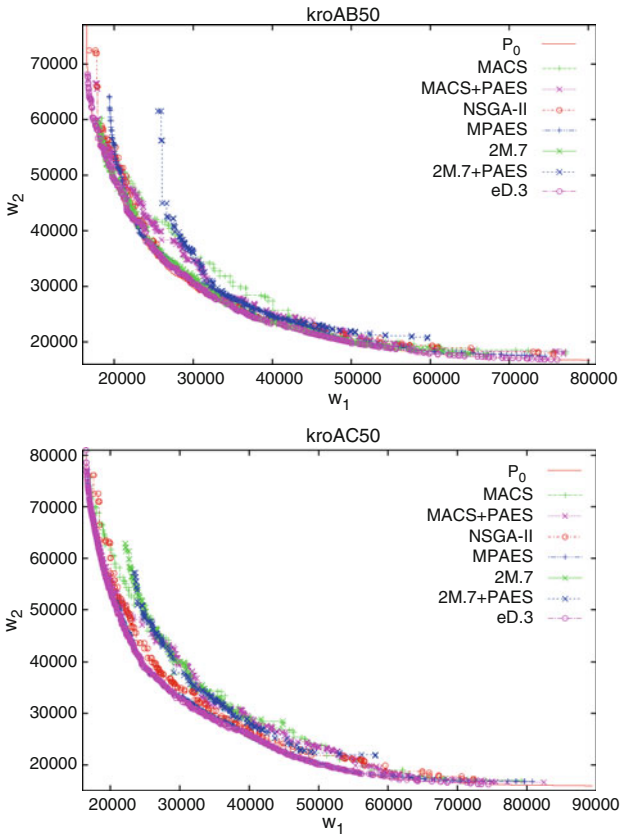

Fig. 12 Typical fronts from the seven methods (MACS, MACS + PAES, NSGA-II, MPAES, 2M.7, 2M.7 + PAES, ϵD.3) over two instances of the MOTSP (kroAB50 and kroAC50). For all instances, the closest method to the reference set ϵD.3

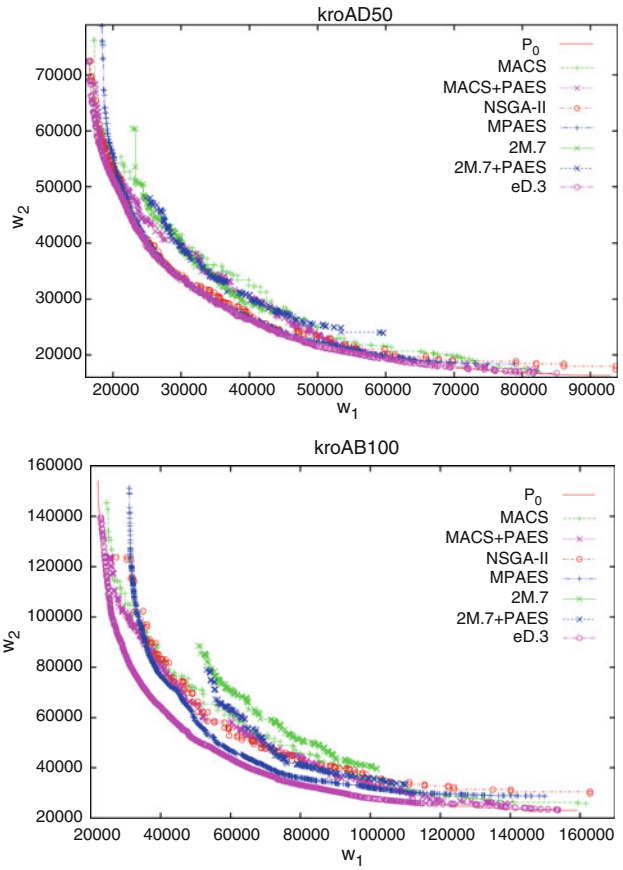

Fig. 13 Typical fronts from the seven methods (MACS, MACS + PAES, NSGA-II, MPAES, 2M.7, 2M.7 + PAES, ϵD.3) over two instances of the MOTSP (kroAD50 and kroAB100). For all instances, the closest method to the reference set ϵD.3

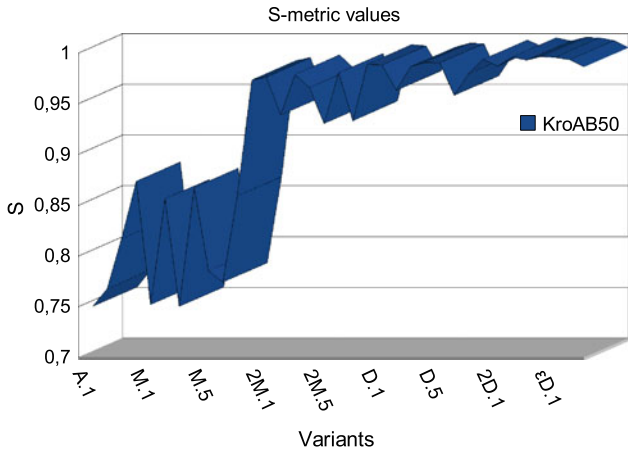

Fig. 14 S-metric values from variants A.1 to ϵD.4 (KroAB50)

- Be as near as possible to the real Pareto front;
- Have spread solutions all along the Pareto front; and
- Be uniformly distributed.

In the next sections four of those metrics are defined: C, S, $R1$, and $R3$.

C-set coverage metric

The *set coverage* metric (Zitzler et al. 2000; Zitzler and Thiele 1999; Deb 2001; Knowles 2002) is computed by

$$\begin{aligned} &C : \Omega^2 \to [0,1] \\ &(\mathcal{P},\mathcal{Q}) \mapsto \frac{|\{q \in \mathcal{Q} : (\exists p \in \mathcal{P} : p \preceq q)\}|}{|\mathcal{Q}|}, \end{aligned} \tag{13}$$

where Ω is the set off all approximation sets, and $\mathcal{P}$ and $\mathcal{Q}$ are two approximation sets. This metric calculates the proportion of elements in $\mathcal{Q}$ that are weakly dominated by at least one element in $\mathcal{P}$.

If $C(\mathcal{P},\mathcal{Q}) = 0$ then none of the elements of $\mathcal{Q}$ is weakly dominated. On the other hand, if $C(\mathcal{P},\mathcal{Q}) = 1$ then all elements of $\mathcal{Q}$ are weakly dominated by at least one of the elements in $\mathcal{P}$. However, this metric cannot determine how much an approximation outperforms another if one of the sets completely dominates the other. Since $C(\mathcal{P},\mathcal{Q}) + C(\mathcal{Q},\mathcal{P})$ is not necessarily equal to 1, both indicators should be considered.

Hyper-volume and $\mathcal{S}$-metric

The hyper-volume metric calculates the hyper-volume of the objective space dominated by an approximation set and an anti-ideal solution (Zitzler 1999). The anti-ideal solution is computed as an objective vector such that its components are the maximum possible value in each objective. If it is not possible to determine the exact anti-ideal solution it should be used one outside the feasible objective space, such that the rectangle defined by the ideal and anti-ideal solution encloses the entire space. However it should be noticed that the use of different points can lead to different results as exampled by Knowles and Corne (2002).

Since the hyper-volume depends on the magnitude of the values, the objectives should be normalized. Alternatively, the normalization can somehow be avoided by computing the *hyper-volume ratio*, that we will simply call $\mathcal{S}$*-metric*, which defined as

$$\begin{aligned} &\mathcal{S} : \Omega \to \mathbb{R}^+ \\ &\mathcal{Q} \mapsto \frac{\mathrm{HV}(\mathcal{Q})}{\mathrm{HV}(\mathcal{P}^*)}, \end{aligned} \tag{14}$$

where $\mathcal{Q}$ is an approximation set, $\mathcal{P}^*$ is the Pareto set (or a reference set), and $\mathrm{HV}(X)$ is the hyper-volume of the region defined by the elements of X and the anti-ideal solution.

This metric has the advantage of measuring both diversity and proximity since values closer to 1 indicate that the approximation set is near to the Pareto set and/or has a higher distribution, all along the Pareto front.

R1 and R3 metrics

This section describes two metrics based on utility functions (Jaszkiewicz 2001; Zitzler et al. 2003). As the name suggest, the comparison indicators based in utility functions employ usefulness functions to compute a value that indicates the worth of the approximation set through that function. Usually, the utility function is a parametric correspondence, and an overall value, for the indicator, can be computed by combining the results when those parameters change in some set.

R1

The $R1$ metric is defined as

$$R1(\mathcal{P}_1,\mathcal{P}_2,U,p) = \int_{u \in U} C(\mathcal{P}_1,\mathcal{P}_2,u)p(u)\,\mathrm{d}u, \tag{15}$$

where $\mathcal{P}_1$ and $\mathcal{P}_2$ are two approximation sets, U is a set of utility functions, $u : \mathbb{R}^m \to \mathbb{R}$, which map each approximation to an utility measure, $p(u)$ is the probability of the utility u, and

$$C(\mathcal{P}_1,\mathcal{P}_2,u) = \begin{cases} 1 & \text{if } u^*(\mathcal{P}_1) < u^*(\mathcal{P}_2) \\ \frac{1}{2} & \text{if } u^*(\mathcal{P}_1) = u^*(\mathcal{P}_2) \\ 0 & \text{if } u^*(\mathcal{P}_1) > u^*(\mathcal{P}_2) \end{cases}, \tag{16}$$

with

$$u^*(\mathcal{P}) = \min_{q \in \mathcal{P}} u(q).$$

The $R1$ metric measures the probability that an approximation set is better than another over the family of utility functions U. To define U, we can use a family of Tchebycheff utility function defined as

$$u_\lambda(q, r) = \max_{j=1,2,\ldots,m} \{\lambda_j(q_j - r_j)\},$$

where $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m)$ is a weight vector, q is a solution for which we want to measure the utility and r is a reference point. Therefore, in formula (15), U is set as

$$U = \left\{ u_\lambda(q, r) : \lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m) \in]0, 1[^m \wedge \sum_{i=1}^{m} \lambda_i = 1 \right\}.$$

If $R1(\mathcal{P}_1, \mathcal{P}_2, U, p) > \frac{1}{2}$ then, according to this measure, $\mathcal{P}_1$ is better than $\mathcal{P}_2$ and it will be not worse if $R1(\mathcal{P}_1, \mathcal{P}_2, U, p) \geq \frac{1}{2}$.

$R1_R$ is defined considering one of the sets as a reference set. For example, if $\mathcal{P}_r$ is the reference set, $R1_R(\mathcal{P}_1, U, p) = R1(\mathcal{P}_r, \mathcal{P}_1, U, p)$ and, therefore, near values of $R1_R(\mathcal{P}_1, U, p)$ to 0.5 indicates that more probably $\mathcal{P}_1$ is a good approximation.

R3

The $R3$ metric is defined as

$$R3(\mathcal{P}_1, \mathcal{P}_2, U, p) = \int_{u \in U} \frac{u^*(\mathcal{P}_1) - u^*(\mathcal{P}_2)}{u^*(\mathcal{P}_1)} p(u)\, \mathrm{d}u, \qquad (17)$$

where $\mathcal{P}_1$,$\mathcal{P}_2$,U, u^*, and p are defined as for Eq. (15).

The $R3$ metric measures the expected proportion of superiority of one set over another.

Similarly to the two previous cases, provided a reference set it is possible to define

$$R3_R(\mathcal{P}_1, U, p) = R3(\mathcal{P}_r, \mathcal{P}_1, U, p),$$

and if $R3_R$ is a value near to 0 then $\mathcal{P}_1$ is expected to be a good approximation.

In practice, the computation of formulas (15) and (17) can be approximated by replacing the integrals by a Riemann sum over U_Λ, where

$$U_\Lambda = \left\{ u_\lambda(q, r) : \lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m) \wedge \lambda_i \in \left\{ \frac{1}{k}, \frac{2}{k}, \ldots, \frac{k-1}{k} \right\} \wedge \sum_{i=1}^{m} \lambda_i = 1 \right\}, \qquad (18)$$

for some large k, that is, those metric values can be approximated by

$$R1(\mathcal{P}_1, \mathcal{P}_2, r, U_\Lambda, p) = \sum_{u_{\lambda,r} \in U_\Lambda} C(\mathcal{P}_1, \mathcal{P}_2, u_{\lambda,r}, r) p(u_{\lambda,r})$$

and

$$R3(\mathcal{P}_1, \mathcal{P}_2, r, U_\Lambda, p) = \sum_{u_{\lambda,r} \in U_\Lambda} \frac{u^*_{\lambda,r}(\mathcal{P}_1) - u^*_{\lambda,r}(\mathcal{P}_2)}{u^*_{\lambda,r}(\mathcal{P}_1)} p(u_{\lambda,r}).$$

Appendix B: Statistical inference

The experimental analysis of an algorithm or algorithms performance is a necessary task to infer about its quality. In some cases, the objective is to decide if a given method is better than another for a given set of problems (since, as the "no-free-lunch" theorem suggests, it is not possible to find an algorithm which is better in every aspect for every problem than all the other algorithms). This leads to the utilization of statistical inference which, given adequate samples, allows to support or reject certain hypothesis. Details about the use of statistical test in the study of results obtained with meta-heuristics can be found in (Demšar 2006; García et al. 2008, 2009; García and Herrera 2008). In particular, in this paper the Mann–Whitney test is used (summarized below).

Mann–Whitney test

The Mann–Whitney test or Wilcoxon rank-sum test (Mann and Whitney 1947) is a nonparametric test to compare non paired groups.

The computation of the Mann–Whitney test starts by ordering the samples from the smaller to their greatest values. To the smallest value is assigned rank one, to the second smallest value is assigned rank two, and so on until the nth value which is ranked with n. In the case of ties, a mean value of the ranks is assigned equal for all those values. Then the ranks from each group are added. The test tries to answer the question: "If the two populations have the same median, which are the chances that a random sampling from the populations would return in a rank sum which is as least as the one observed by the experiment?".

To answer this question, the rank sum is used to obtain a p value from an appropriated table. If p is small we can reject the idea that the obtained difference is a coincidence and conclude that the populations have distinct medians. If p is large, the data does not allow us to conclude that the medians are different, which is not the same to say that they are equal.

Before applying the Mann–Whitney test we should verify some premises, as for example: the observed values

are independent, the observations are not paired (if not, think of applying the Wilcoxon signed-rank test), we really want to compare medians, and the samples don't come from Gaussian populations (for which other, more powerful, test can be applied).

References

Aarts E, Lenstra J (1997) Local search in combinatorial optimization. Wiley-Interscience Series in discrete mathematics and optimization. Wiley, London

Barán B, Schaerer M (2003) A multiobjective ant colony system for vehicle routing problem with time windows. In: 21st IASTED International Multi-Conference on Applied Informatics, Innsbruck, Austria, pp 97–102

Blum C (2005a) Ant colony optimization: introduction and recent trends. Phys Life Rev 2(4):353–373

Blum C (2005b) Beam–ACO—hybridizing ant colony optimization with beam search: an application to open shop scheduling. Comput Oper Res 32(6):1565–1591

Blum C, Roli A (2003) Metaheuristics in combinatorial optimization: overview and conceptual comparison. ACM Comput Surv (CSUR) 35(3):268–308

Blum C, Bautista J, Pereira J (2006) Beam–ACO applied to assembly line balancing. LNCS 4150:96–107. doi:10.1007/11839088

Bui L, Essam D, Abbass H, Green D (2001) Performance analysis of evolutionary multiobjective optimization methods in noisy environments. Tech. Rep. TR-ALAR-200504006, University of New South Wales, Australia

Camerini P, Galbiati G, Maffioli F (1980) Complexity of spanning tree problems: part I. Eur J Oper Res 5:346–352

Camerini P, Galbiati G, Maffioli F (1983) On the complexity of finding multi-constrained spanning trees. Discret Appl Math 5:39–50

Deb K (2001) Multi-objective optimization using evolutionary algorithms. Wiley, London

Deb K, Pratap A, Agarwal S, Meyarivan T (2000) A fast elitist multi-objective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 6:182–197

Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7:1–30

Dorigo M, Stützle T (2004) Ant colony optimization. MIT Press, Cambridge

Dorigo M, Bonabeau E, Theraulaz G (1999) Swarm intelligence: from natural to artificial systems. Oxford University Press, Oxford

Ehrgott M, Gandibleux X (2004) Approximative solution methods for multiobjective combinatorial optimization. TOP 12(1):1–63

Gambardella L, Dorigo M (2000) An ant colony system hybridized with a new local search for the sequential ordering problem. INFORMS J Comput 12(3)

Gambardella LM, Taillard E, Agazzi G (1999) Macs-vrptw: a multiple ant colony system for vehicle routing problems with time windows. New ideas in optimization, pp 63–76

García S, Herrera F (2008) Design of experiments in computational intelligence: on the use of statistical inference. In: HAIS '08: Proceedings of the 3rd international workshop on hybrid artificial intelligence systems. Springer, Berlin, pp 4–14. doi:10.1007/978-3-540-87656-4_3

García S, Molina D, Lozano M, Herrera F (2009) A study on the use of non-parametric tests for analyzing the evolutionary algorithms behaviour: a case study on the CEC2005 special session on real parameter optimization. J Heuristics 15(6):617–644

García S, Fernandez A, Luengo J, Herrera F (2009) A study of statistical techniques and performance measures for genetics-based machine learning: accuracy and interpretability. Soft Comput 13(10):959–977

García-Martínez C, Cordón O, Herrera F (2004) An empirical analysis of multiple objective ant colony optimization algorithms for the bi-criteria TSP. ANTS Workshop, Lecture Notes in Computer Science 3172:61–72

García-Martínez C, Cordón O, Herrera F (2007) A taxonomy and an empirical analysis of multiple objective ant colony optimization algorithms for the bi-criteria TSP. Eur J Oper Res 127(1):116–148

Garey M, Johnson D (1990) Computers and intractability; a guide to the theory of NP-Completeness. W. Freeman & Co., New York

Glover F, Laguna M (1997) Tabu search. Kluwer, Dordrecht

Hamacher H, Ruhe G (1994) On spanning tree problems with multiple objective. Ann Oper Res 52:209–230

Ishibuchi H, Murata T (1998) Multi-objective genetic local search algorithm and its application to flowshop scheduling. IEEE Trans Syst Man Cybern 3(28):392–403

Jaszkiewicz A (2001) Multiple objective metaheuristic algorithms for combinatorial optimization. PhD thesis, Poznan University of Technology

Jaszkiewicz A (2002) Genetic local search for multiple objective combinatorial optimization. Eur J Oper Res 137:50–71

Jaszkiewicz A (2006) A comparative experiment with multiple objective genetic local search algorithm on multi-objective travelling salesperson problem. http://www-idss.cs.put.poznan.pl/~jaszkiewicz/motsp

Johnson S (2001) Emergence. The connected lives of ants, brains, cities, and software. Scribner

Kampstra P (2008) Beanplot: a boxplot alternative for visual comparison of distributions. J Stat Softw Code Snippets 28(1):1–9

Kirkpatrick S, Gelatt C, Vecchi M (1983) Optimization by simulated annealing. Science 220(4598):671–680

Knowles J (2002) Local-search and hybrid evolutionary algorithms for pareto optimization. PhD thesis, University of Reading, UK

Knowles J, Corne D (2000a) Approximating the nondominated front using the pareto archived evolution strategy. Evol Comput MIT Press 8(22):149–172

Knowles J, Corne D (2000b) M-PAES: a memetic algorithm for multiobjective optimization. In: Proceedings of the 2000 Congress on evolutionary computation, vol 1. IEEE Press, La Jolla, pp 325–332

Knowles J, Corne D (2001) A comparative assessment of memetic, evolutionary, and constructive algorithms for the multiobjective *d*-MST problem. In: 2nd Workshop on memetic algorithms, WOMA2001, pp 162–167

Knowles J, Corne D (2002) On metrics for comparing non-dominated sets. In: Congress on evolutionary computation (CEC 2002), vol 1. pp 711–716. doi:10.1109/CEC.2002.1007013

Levine J, Ducatelle F (2004) Ant colony optimisation and local search for bin packing and cutting stock problems. J Oper Res Soc Special Issue on Local Search 55(7):705–716

Mann H, Whitney D (1947) On a test of whether one of two random variables is stochastically larger than the other. Ann Math Stat 18:50–60

Middendorf M, Reischle F, Schmeck H (2002) Multi colony ant algorithms. J Heuristics 8:305–320

Miettinen K (1999) Nonlinear multiobjective optimization. Kluwer, Dordrecht

Murata T (1997) Genetic algortithms for multi-objective optimization. PhD thesis, Osaka Prefecture University

Paquete L, Stützle T (2003) A two-phase local search for the biobjective traveling salesman problem. In: Fonseca C, Fleming

P, Zitzler E, Deb K, Thiele L (eds) Second International Conference on evolutionary multi-Criterion optimization, EMO 2003, vol 2632. Faro, Portugal, pp 479–493

Paquete L, Stützle T (2006) A study of stochastic local search algorithms for the biobjective QAP with correlated flow matrices. Eur J Oper Res 169(3):943–959

Parmee I (2001) Evolutionary and adaptive computing in engineering design. Springer, London, iSNB:1-85233-029-5

Reimann M, Laumanns M (2004) A hybrid aco algorithm for the capacitated minimum spanning tree problem. In: Blum C, Roli A, Sampels M (eds) First International Workshop on hybrid metaheuristics (HM 2004). Valencia, Spain, pp 1–10

Reimann M, Doerner K, Hartl R (2004) D-ants: savings based ants divide and conquer the vehicle routing problem. Comput Oper Res 31(4):563–591

Romero C (1993) Teorí a de la decisión multicritério: conceptos, técnicas y aplicaciones (in spanish). Alianza Universidad Textos

Talbi EG (2002) A taxonomy of hybrid metaheuristics. J Heuristics 8(5):541–564

Ulungu E, Teghem J, Fortemps P, Tuyttens D (1999) MOSA method: a tool for solving multiobjective combinatorial optimization problems. J Multi-Criteria Decis Anal 8(4):221 – 236

Zitzler E (1999) Evolutionary algorithms for multiobjective optimization: methods and applications. PhD thesis, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland

Zitzler E, Thiele L (1999) Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. IEEE Trans Evol Comput 3(4):257–271

Zitzler E, Deb K, Thiele L (2000) Comparison of multiobjective evolutionary algorithms: empirical results. Evol Comput 8:173–195

Zitzler E, Thiele L, Laumanns M, Fonseca C, Fonseca V (2002) Why quality assessment of multiobjective optimizers is difficult. In: Proceedings of the genetic and evolutionary computation conference (GECCO 2002), pp 666–674

Zitzler E, Thiele L, Laumanns M, Fonseca C, Fonseca V (2003) Performance assessment of multiobjective optimizers: an analysis and review. IEEE Trans Evol Comput 7(2):117–132