

Non-dominated sorting differential evolution algorithm for the minimization of route based fuel consumption multiobjective vehicle routing problems

Iraklis-Dimitrios Psychas¹ · Magdalene Marinaki¹ ·
Yannis Marinakis¹ · Athanasios Migdalas^{2,3}

Received: 22 September 2015 / Accepted: 5 May 2016 / Published online: 18 May 2016
© Springer-Verlag Berlin Heidelberg 2016

Abstract In this paper, three parallel multi-start non-dominated sorting differential evolution algorithms (PMS-NSDEs) are proposed for the solution of four multiobjective route based fuel consumption vehicle routing problems (MRFCVRPs) and their results are compared with the results of a parallel multi-start NSGA II algorithm. All these algorithms use more than one initial population of solutions. In each algorithm a variable neighborhood search algorithm for the improvement of each solution separately is used. The problems that are formulated with two competitive objective functions are the multiobjective symmetric and asymmetric delivery route based fuel consumption vehicle routing problem (MSDRFCVRP and MADRFCVRP) and the multiobjective symmetric and asymmetric pick-up route based fuel consumption vehicle routing problem (MSPRFCVRP and MAPRFCVRP). The objective functions correspond to the optimization of the time needed for the vehicle to travel between two customers or between the customer and the depot and to the route based fuel consumption of the vehicle considering the traveled distance, the load of the vehicle, the slope of the road, the speed and the direction of the wind, and the driver's behavior when the decision maker plans delivery or pick-up routes. A number of modified Vehicle Routing Problem instances are used in order to measure the quality of the proposed algorithms.

✉ Yannis Marinakis
marinakis@ergasya.tuc.gr

¹ School of Production Engineering and Management, Technical University of Crete, 73100 Chania, Greece

² Luleå University of Technology, Industrial Logistics, 97187 Luleå, Sweden

³ Department of Civil Engineering, Aristotle University of Thessalonike, 54124 Thessaloniki, Greece

Keywords Multiobjective route based fuel consumption vehicle routing problem · Parallel multi-start NSGA II · Parallel multi-start NSDE · Variable neighborhood search

1 Introduction

In real world applications, most of the optimization problems have more than one objective functions for optimization. Thus, one way to solve these problems (besides the multicriteria optimization, the hierarchical optimization, the goal programming and so on) is the use of an evolutionary multiobjective approach where the finding of the Pareto Front of the non-dominated solutions is needed [13]. One of the problems in the supply chain management that could be formulated as a multiobjective optimization problem is the vehicle routing problem (VRP) as a number of different objectives functions could be used. The VRP belongs to the class of NP-hard optimization problems [34]. For an overview of the VRP please see [45, 77, 78]. In real world applications in order to prove that the quality of the routes is good enough usually more than one criterion are optimized. Thus, in recent years a large number of Multiobjective Vehicle Routing problems have been published [36]. The multiobjective vehicle routing problem (moVRP) is a VRP problem where simultaneous optimization of more than one objective functions is required. For more information about moVRPs please see [43].

In recent years, the optimization of energy reduction or fuel consumption in the Vehicle Routing Problem and other problems has been studied. The calculation of the tonne-kilometers (tonne-km or tkm) of the vehicle taking into account the covering distance and the load of the vehicle could be an easy way to calculate the “Fuel Efficiency” and the “CO₂ emissions” of a vehicle [37, 55, 56]. Considering the Leonardi’s et al. research [47] the “Efficiency of the vehicle use” can be calculated by a ratio tonne-kilometres/mass-kilometres and in order to calculate the “CO₂ Efficiency” they assume that there are some other real route and environmental parameters that are multiplied with the “Efficiency of the vehicle use” in order to calculate the “CO₂ Efficiency” of a vehicle. In order to calculate the mass-kilometres, the weight of the empty vehicle is added to the load of the vehicle. Another parameter that can be taken into account for the calculation of fuel consumption is the parameter of speed [5, 6, 25, 42, 49, 74]. Xiao et al. [80] propose the fuel consumption rate (FCR) for a VRP (FCVRP) in order to minimize the fuel consumption using the distance traveled and the load. The FCR and the “CO₂ emission rate” (CER) were used by Zhang et al. [83] in order to calculate the amount of CO₂ emissions of a vehicle. A bi-objective green vehicle routing problem was proposed in [33] in order to minimize the total traveled distance and the CO₂ emissions. In this research it is referred that in real world problems the vehicle’s load, the vehicle’s speed, the weather conditions (headwinds or backwinds) and the traffic congestion are factors that could affect the fuel economy of a vehicle. The slope of the road (measured in grades between 0 and 10 %) could affect the CO₂ emissions of a vehicle [12, 79]. In [17] a bi-objective pollution routing problem’s model is proposed where the first objective function minimizes the CO₂ emissions of a vehicle and the second objective function minimizes the driving time. For two more complicated multiobjective Energy VRPs please see [41, 59]. Two

other energy Pick-up and Delivery VRP models are analyzed in [50, 75]. Some other CO_2 emissions minimizing models are presented in [10, 23, 38, 39, 48, 76]. Also, CO_2 emissions could be minimized by creating shortest routes and by traveling with the best speed for the environment [70]. For a more extended review for the energy and green vehicle routing problems please see [44, 51].

In this paper, three parallel multi-start non-dominated differential evolution algorithms (PMS-NSDEs) are used for the solution of the proposed problems. In order to apply the differential evolution algorithm [63, 73] in multiobjective optimization problems the single objective algorithm has to be modified in such a way in order to calculate the set of non-dominated solutions. The two main issues that they should be solved in a multiobjective differential evolution algorithm are how the diversity of the solutions will be maintained and how to retain and select the best individuals [8, 57]. A number of multiobjective differential evolution algorithms have been applied mainly in continuous optimization problems and in some cases in combinatorial optimization problems [1–4, 9, 32, 40, 62, 66, 67, 71, 81, 82].

Although the differential evolution algorithm is suitable for continuous optimization problems, in the last years a number of publications concerning the application of the differential evolution algorithm in single and multiobjective routing problems have been produced. In 2009, Erbao and Mingyong [20] proposed a hybrid intelligent algorithm based on the Differential Evolution algorithm in order to solve the vehicle routing problem with fuzzy demands. In the next year (2010), the same group of authors, Mingyong and Erbao [58] using an intelligent DE algorithm (IDE) solved a vehicle routing problem with simultaneous pickups and deliveries and time windows and an open vehicle routing problem with fuzzy demands [21]. Three years later Silva et al. [72] solved the capacitated vehicle routing problem using four discrete DE algorithms. In 2014, Erbao et al. [22] solved an open vehicle routing problem with uncertain demands using the IDE algorithm but in that case it is not necessary for the vehicle to return to its initial location after the service of the costumers. In 2015, Marinakis et al. [54] solved a vehicle routing problem with stochastic demands and a probabilistic traveling salesman problem using differential evolution algorithm. Variants of DE algorithm have been used for the solution of the traveling salesman problem (TSP). In [69], Sauer proposed a discrete DE algorithm using two different local search methods for the solution of the TSP while in 2015, Psychas et al. [64] proposed two variants of the DE algorithm in order to solve the multiobjective TSP.

In this paper, for the solution of the multiobjective vehicle routing problems an improved version of the method published in [64] for the solution of the multiobjective traveling salesman problem is presented. In [64], three new trial vector equations have been presented and the produced algorithm, denoted as non dominated sorting differential evolution algorithm (NSDE), used the rank and the crowding distance to sort the solutions of the parents and the offspring of each iteration. The results of the algorithm (in the multiobjective traveling salesman problem) were compared with the results of other multiobjective differential evolution algorithms (MODE) in order to give the efficiency of the proposed methodology. In this version of the algorithm, denoted as parallel multi-start non-dominated differential evolution algorithm (PMS-NSDE), a number of common characteristics with the previous version exist (the three trial vector equations) but also there are four basic differences (improvements)

of the proposed algorithm. In the proposed algorithm, more than one different initial populations are created and the step of the algorithm that sorts the parents and the offspring according to the rank and the crowding distance is in a different phase of the algorithm. Also, the VNS method that is used in the proposed research is the same with the one used in [65] and is much more sophisticated than the VNS method that was used at [64]. Finally, in the proposed algorithm there is an extra step that improves the best solutions in every iteration using the proposed VNS method [65]. Also, for this research in order to test the efficiency of the proposed algorithms the parallel multi-start NSGA II (PMS-NSGA II) algorithm is used and is compared with them. PMS-NSGA II is an improved version of NSGA II (non-dominated sorting genetic algorithm) [15, 16] and was originally proposed in [65]. A number of variants of the NSGA II algorithm have been used in the literature for solving multiobjective Vehicle Routing Problems, e.g. for solving VRP with route balancing [35], for solving multiobjective VRP problems with time windows [30] and for solving a green vehicle routing problem [33]. Multiobjective Genetic Algorithms for the solution of Multiobjective Vehicle Routing Problems have been used in [7, 61].

The structure of the paper is as follows. In Sect. 2, the optimization models of the MRFCVRPs are described. In Sect. 3, the proposed parallel multi-start algorithms are presented and analyzed. In Sect. 4, the evaluation measures used in the comparisons are presented. In Sect. 5, the computational results are presented. In Sect. 6, a method to decide the most suitable for a decision maker solution of a Pareto front is proposed and, finally, concluding remarks and the future research are given in the last section.

2 Multiobjective route based fuel consumption vehicle routing problems

In this paper, four multiobjective route-based fuel consumption vehicle routing problems (MRFCVRPs) are analyzed and formulated using three different objective functions that are given in the following. Each of the four problems is a 2-objective optimization problem. The vehicle routing problems analyzed in this research differ between them. Two of them are delivery problems and the other two are pick-up problems. Thus, from the three objective functions presented in this paper the first two are used for the delivery problems while the first and the third objective functions are used for the pick-up problems. This formulation is an improvement of the formulation that was presented in [65] where the multiobjective energy reduction vehicle routing problem was formulated and solved.

For all multiobjective route-based fuel consumption vehicle routing problems (MRFCVRPs) studied in this paper, the first objective function is common for the delivery and the pick-up problems and is used for the minimization of the time needed to travel between two customers or a customer and the depot. Thus, if t_{ij}^{κ} is the time needed to visit customer j immediately after customer i using vehicle κ and s_j^{κ} is the service time of customer j using vehicle κ , then, the first objective function is [65]:

$$\min OF1 = \sum_{i=1}^n \sum_{j=1}^n \sum_{\kappa=1}^m (t_{ij}^{\kappa} + s_j^{\kappa}) x_{ij}^{\kappa} \quad (1)$$

Table 1 Grade Index and Beaufort Index

Grade Index (G_i)		Beaufort Index (B_i)	
Uphill	Downhill	Head-wind	Back-wind
$1+(\text{Grade}/10 \%)$	$1-(\text{Grade}/10 \%)$	$1+(\text{Beaufort}/12)$	$1-(\text{Beaufort}/12)$

where n is the number of nodes and m is the number of homogeneous vehicles and the depot is denoted by $i = j = 1$.

The second objective function is used for the minimization of the route based fuel consumption (RFC) taking into account, also, real life route parameters (weather conditions or uphill and downhill or driver’s behavior) [33,47] in addition to the load and the traveled distance when the vehicle travels between two customers or a customer and the depot in the case that the vehicle performs only deliveries in its route. The vehicle should begin with full load and after a visitation to a customer the load is reduced based on the demand of the customer. If we consider that the more loaded is the vehicle the more fuel it consumes, we take the following objective function:

$$\min OF2 = \sum_{j=1}^n \sum_{\kappa=1}^m c_{1j} x_{1j}^{\kappa} \left(1 + \frac{y_{1j}^{\kappa}}{Q} \right) r_{1j} + \sum_{i=2}^n \sum_{j=1}^n \sum_{\kappa=1}^m c_{ij} x_{ij}^{\kappa} \left(1 + \frac{y_{i-1,i}^{\kappa} - D_i}{Q} \right) r_{ij} \tag{2}$$

with the maximum capacity of the vehicle denoted by Q , the i customer has demand equal to D_i and $D_1 = 0$, x_{ij}^{κ} denotes that the vehicle κ visits customer j immediately after customer i with load y_{ij}^{κ} and $y_{1j}^{\kappa} = \sum_{i=1}^n D_i$ for all vehicles as the vehicle begins with load equal to the summation of the demands of all customers assigned in its route and c_{ij} is the distance from node i to node j . The parameter r_{ij} corresponds to the route parameters from the node i to the node j and it is always a positive number. Due to the fact that it may be $r_{ij} \neq r_{ji}$ the product $c_{ij}r_{ij}$ leads to an asymmetric formulation of the whole problem. If the values of r_{ij} is lower than 1 we consider that the route from i to j is a downhill or the wind is back-wind or the driver drives with smooth shifting. If r_{ij} is larger than 1 we consider that the route from i to j is an uphill or the wind is a head-wind or the driver drives with aggressive shifting. If the $r_{ij} = 1 \forall (i, j)$ that belongs to the route, then, the problem is a symmetric problem.

In order to calculate the r_{ij} parameter in real life problems the following method is used: Based on Cicero et al. [12] the roads where the vehicles can travel safely have slope in grades between 0 and 10 %. Also, the beaufort scale that is used for the measurement of the wind speed consists of an integer number between 0 and 12 [14]. In the proposed model, the Grade Index (G_{ij}) and the Beaufort Index (B_{ij}) are calculated in two different ways considering the road (if it is uphill or downhill) and if the wind is head-wind or back-wind taking into account Table 1.

Considering the G_{ij} and the B_{ij} , the r_{ij} parameter is calculated as follows:

$$r_{ij} = \frac{G_{ij} + B_{ij}}{2} \tag{3}$$

The third objective function is used for the minimization of the route based fuel consumption (*RFC*) in the case that the vehicle performs only pick-ups in its route. The vehicle should begin with empty load and after a visitation to a customer the load is increased based on the pick-up amount of the customer. If we consider that the more loaded is the vehicle the more fuel it consumes, we take the following objective function:

$$\min OF3 = \sum_{j=1}^n \sum_{\kappa=1}^m c_{1j} x_{1j}^{\kappa} r_{1j} + \sum_{i=2}^n \sum_{j=1}^n \sum_{\kappa=1}^m c_{ij} x_{ij}^{\kappa} \left(1 + \frac{y_{i-1,i}^{\kappa} + D_i}{Q} \right) r_{ij} \quad (4)$$

with r_{ij} is the route parameters as in the OF2 and $y_{1j}^{\kappa} = 0$ for all vehicles as the vehicle begins with empty load. In that case the D_i is the pick-up amount of the customer i .

The constraints of the problems are the following:

$$\sum_{j=1}^n \sum_{\kappa=1}^m x_{ij}^{\kappa} = 1, \quad i = 1, \dots, n \quad (5)$$

$$\sum_{i=1}^n \sum_{\kappa=1}^m x_{ij}^{\kappa} = 1, \quad j = 1, \dots, n \quad (6)$$

$$\sum_{j=1}^n x_{ij}^{\kappa} - \sum_{j=1}^n x_{ji}^{\kappa} = 0, \quad i = 1, \dots, n, \quad \kappa = 1, \dots, m \quad (7)$$

$$\sum_{j=1, j \neq i}^n y_{ji}^{\kappa} - \sum_{j=1, j \neq i}^n y_{ij}^{\kappa} = D_i, \quad i = 1, \dots, n, \quad \kappa = 1, \dots, m, \text{ for deliveries} \quad (8)$$

$$\sum_{j=1, j \neq i}^n y_{ij}^{\kappa} - \sum_{j=1, j \neq i}^n y_{ji}^{\kappa} = D_i, \quad i = 1, \dots, n, \quad \kappa = 1, \dots, m, \text{ for pick-ups} \quad (9)$$

$$Q x_{ij}^{\kappa} \geq y_{ij}^{\kappa}, \quad i, j = 1, \dots, n, \quad \kappa = 1, \dots, m \quad (10)$$

$$x_{ij}^{\kappa} = \begin{cases} 1, & \text{if } (i, j) \text{ belongs to the route} \\ 0, & \text{otherwise} \end{cases} \quad (11)$$

Constraints (5) and (6) represent that each customer must be visited only by one vehicle; constraints (7) ensure that each vehicle that arrives at a node must leave from that node also. Constraints (8) and (9) indicate that the reduced (if it concerns deliveries) or increased (if it concerns pick-ups) load (cargo) of the vehicle after it visits a node is equal to the demand of that node. Constraints (10) are used to limit the maximum load carried by the vehicle and to force y_{ij}^{κ} to be equal to zero when $x_{ij}^{\kappa} = 0$ while constraints (11) ensure that only one vehicle will visit each customer.

Another differentiation of the problems is when a symmetric or an asymmetric case is studied. When the multiobjective symmetric delivery route-based fuel consumption vehicle routing problem (MSDRFCVRP) is studied, the combination of OF1 and

OF2 is required considering that $r_{ij} = 1 \forall (i, j)$ that belongs to the route while when the multiobjective asymmetric delivery route-based fuel consumption vehicle routing problem (MADRFCVRP) is studied the combination of OF1 and OF2 is required considering that $r_{ij} \neq r_{ji} \forall (i, j)$ that belongs to the route. The same holds when a problem with pick-ups is solved. More precisely, when the multiobjective symmetric pick-up route-based fuel consumption vehicle routing problem (MSPRFCVRP) is studied, the combination of OF1 and OF3 is required considering that $r_{ij} = 1 \forall (i, j)$ that belongs to the route while when the multiobjective asymmetric pick-up route-based fuel consumption vehicle routing problem (MAPRFCVRP) is solved, the combination of OF1 and OF3 is required considering that $r_{ij} \neq r_{ji} \forall (i, j)$ that belongs to the route.

3 Parallel multi-start algorithms

In this Section, all the proposed Multiobjective algorithms are presented. The common steps of all algorithms are the solutions' representation, the method using for producing the initial solutions and the variable neighborhood search (VNS) algorithm (in order to increase the exploitation abilities of each solution) that were presented and analyzed in detail at [65].

For all the algorithms, the solutions are represented with the path representation of the tour. Each solution (individual for the differential evolution algorithm) is represented by a n -dimensional vector in problem space and its performance is evaluated on the predefined fitness functions. For example, if we have a solution with five nodes a possible path representation would be the "1 2 3 4 5". If these routes do not start with node 1 (the depot), we find it and put it at the beginning of the route. As we would like the size of the solution's vector to be maintained constant for all members of the population and for all the procedures in an iteration and, finally, for all iterations we use a path representation in the solutions without using in the solution's vector the return to the depots. The solution vector is transformed in the form of "1 2 3 1 4 5 1" with the addition of the depots only when the objective functions of the problem are calculated considering the constraints of the problem in order to find afterwards from which node the vehicle will return to the depot. Also, for all the algorithms we assume that we have X different *initial populations* with W solutions for each population.

Each population is separated in K sub-populations of w solutions, where K is the number of objective functions ($w = W/K$). For the calculation of the first solution of each sub-population, a different method is applied (VNS algorithm [29,65], the Nearest Neighborhood method [46] and a variant of GRASP method [24]). The variant of GRASP algorithm works as follows: In this GRASP algorithm instead of using the restricted candidate list (RCL), we use the following procedure: Always node 1 is used as a starting node. Then, a random number ($Rand$) equal to 0 or 1 is generated. If $Rand = 0$, then, the nearest node to a node i is visited. If $Rand = 1$, the second nearest node to a node i is visited. For the calculation of the rest solutions of each sub-population a Swap method [46], the 2-opt method [52] and a random method are used. The variant of variable neighborhood search (VNS) works as follows: initially, the 2-opt local search algorithm [46] is applied for a specific number of iterations ($local_{max}$). For the proposed algorithm, the vns_{max} and the $local_{max}$ were set equal to

10 in order not to increase the computational time of the algorithm. If 2-opt improves the solution (the new solution dominates the old solution), then, 2-opt algorithm is applied for $local_{max}$ number of iterations. On the other hand, if 2-opt is trapped in a local optimum (the new solution is dominated by the old solution or the two solutions are non-dominated between them) when $local_{max}$ number of iterations has been reached, the 3-opt algorithm [46] is applied. The same procedure is followed when the current local search algorithm is trapped in a local optimum, by replacing the current algorithm initially with the Swap algorithm, afterwards with the 2-2 exchange algorithm, then, with the 1-0 relocate and finally, with the 2-0 relocate algorithms [46]. For more information please see [65].

3.1 Parallel multi-start non-dominated sorting differential evolution algorithms (PMS-NSDEs)

After the calculation of the initial population, the main phase of the non-dominated sorting differential evolution method is applied. The single objective differential evolution algorithm works as follows. Initially, the mutation operator produces a trial vector $u_i(it)$ for each individual $x_j(it)$ of the current population by mutating a target vector with a weighted differential. The trial vector, $u_i(it)$, is generated as follows: a target vector, $x_{i_1}(it)$, is selected from the population, such that $i \neq i_1$. Then, two individuals, x_{i_2} and x_{i_3} , are selected randomly from the population such that $i \neq i_1 \neq i_2 \neq i_3$ [73]. Using these individuals, the trial vector is calculated by perturbing the target vector as follows:

$$u_i(it) = x_{i_1}(it) + \beta(x_{i_2}(it) - x_{i_3}(it)) \quad (12)$$

where $\beta \in (0, \infty)$ is the scale factor. The upper bound of β is usually the value 1 because as it has been proved if the $\beta > 1$ there is no improvement in the solutions [19,63].

After the completion of the mutation phase of the algorithm a crossover operator Cr is, usually, applied. Then, depending on the comparison of the Cr value with a random number for each element of the solution vector, $rand_i(0, 1)$, the corresponding value is inherited either from the trial vector or from the parent. In the next generation the fitness function of the parent and of the offspring are compared and the best one survives [19,63,73].

The first issue that we have to solve in the algorithm was that the solutions for the problem are represented via the path representation of the tours but this representation is not suitable for a DE algorithm. Thus, for the calculation of each trial vector the solutions are transformed into a floating point in the interval [0,1] and after the calculation of the trial vector the solutions are transformed back into integer points creating new routes [53,65].

In the proposed algorithms, the procedure starts with the mutation operator for the production of a trial vector for each individual of the current population by mutating a target vector with a weighted differential. In this research, three mutation operators are applied. All equations are the same with the ones used in [64]:

$$u_i(it) = x_{i_1}(it) + \beta(x_{i_2}(it) - x_{i_3}(it)) \quad (13)$$

$$u_i(it) = x_{i_1}(it) + \beta(Pareto_{i_2}(it) - Pareto_{i_3}(it)) \quad (14)$$

$$u_i(it) = Pareto_{i_1}(it) + \beta(x_{i_2}(it) - x_{i_3}(it)) \quad (15)$$

where the solutions x_i are random solutions from the population and the solutions $Pareto_i$ are random solutions from the Pareto front. In the proposed algorithms, the crossover operator is not used as it was observed after a large number of tests using different values for C_r that the results were the best when the C_r is set equal to 1. The only difference between the three algorithms are these equations and, thus, the three different produced algorithms are denoted as PMS-NSDE1, PMS-NSDE2 and PMS-NSDE3, using Eqs. 13, 14 and 15, respectively.

The calculation of the trial vectors could produce some inefficient (due to the transformation of the solutions from continuous values (suitable for the equations of DE) to discrete values (path representation) and vice versa) and dominated from the parents's solutions of the previous iteration. Thus, it was decided to add another phase for the calculation of the new parents in the algorithm in order to take advantage of possible good new and old solutions in the whole set of individuals. Thus, after the calculation of the trial vectors the parents-individuals $x_i(it)$ and the trial vectors $u_i(it)$ are combined in a new vector and, then, the members of the new vector are sorted using the *rank* and the *crowding distance* as in the NSGA II algorithm [65]. The first W individuals of the new vector are the produced solutions of the iteration it . With this procedure we give the opportunity to the solutions of the previous iteration that are dominated from the new solutions to be improved in the current iteration and we avoid to add to the next iterations inefficient solutions that will probably be produced using the equations of the trial vector. At the next step the Variable Neighborhood Search (VNS) algorithm is applied to the individuals with both the $vnsm_{max}$ and the $local_{max}$ equal to 10 [65].

The new solution of each individual is found by using the following observation. If the offspring in iteration it dominates its parent of the iteration $it - 1$, then, the parent is replaced by the offspring. On the other hand, if the parent dominates the offspring, then, the parent remains in the population. Finally, if these two solutions are not dominated between them, then, the parent remains in the population. This is performed as it is desirable to give to the individual more exploration abilities. It should be noted that the non-dominated solutions are not deleted from the Pareto front and, thus, the good solutions will not disappeared from the population. Finally, in order to improve the solutions the VNS method is applied at the best solutions with both the $vnsm_{max}$ and the $local_{max}$ equal to 10 [65]. In the next iterations in order to insert an individual in the Pareto front archive there are two possibilities. First, the individual is non-dominated with respect to the contents of the archive and second it dominates any individuals in the archive, but in this case all the dominated individuals have to be deleted from the archive. At the end of each iteration, from the non-dominated solutions from all initial populations the Total Pareto Front is updated considering the non-dominated solutions of the last *initial population*. A pseudocode of the algorithm is the following:

Do while the maximum number of the Populations has not been reached:

Initialization

Selection of the number of individuals

Generation of the initial population

Evaluation of the population for each objective function

Selection of the β

Initialization of the Population's Pareto Front

Main Phase

Do while the maximum number of generations has not been reached:

For every individual

Calculate the trial vector

Evaluation of the individuals trial vector for each objective function

Endfor

Add the individuals and the produced trial vectors in a new vector

Calculate the *rank* and the *crowding distance* of the individuals and the produced trial vector

Sort them according to *rank* and *crowding distance*

Select the first W individuals

Application of VNS on each individuals

Update the best solution of each individual

Update of the Pareto front

Enddo

Return Population's Pareto Front

Enddo

Return Total Pareto Front

In order to test the efficiency of the proposed parallel multi-start non-dominated sorting differential evolution algorithms, a parallel multi-start NSGA II algorithm [65] is used to compare with the proposed algorithms.

4 Evaluation measures

In this paper, as the optimum Pareto Front is not known, four different evaluation measures are used for the comparison of the Pareto Fronts of the four algorithms [65]:

- The range to which the front spreads out is described by the following equation [84]:

$$M_k = \sqrt{\sum_{i=1}^K \max\{\|p' - q'\|\}} \quad (16)$$

where K is the number of objectives and p', q' are the values of the objective functions of two solutions that belong to the Pareto front.

- The number of solutions of the Pareto front (L).

- The Δ measure includes information about both spread and distribution of each solutions [60]. For the calculation of the Δ measure the following equation is used:

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{|S|-1} |dist_i - \overline{dist}|}{d_f + d_l + (|S| - 1)\overline{dist}} \tag{17}$$

where S is the number of the intermediate solutions between the extreme solutions, d_f and d_l are the Euclidean distances between the extreme solutions [60] and the boundary solutions [60] of the obtained non-dominated set [16], $dist_i$ is the distance from a boundary solution i to the next boundary solution, $i = 1, 2, \dots, (S - 1)$ and \overline{dist} is the average value of all $dist_i$ distances.

- Coverage [68, 84]: for a pair (A_1, B_1) of approximation sets of Pareto solutions of two different algorithms the fraction of solutions in B_1 that are weakly dominated by one or more solutions in A_1 . The coverage measure (C measure) is calculated by the following equation:

$$C(A_1, B_1) = \frac{|\{b \in B_1; \exists a \in A_1 : a \leq b\}|}{|B_1|} \tag{18}$$

5 Computational results

The whole algorithmic approach, which was implemented in Visual C++, was tested on a set of instances. As it was mentioned previously, in the multiobjective (K -objective) VRP, K different objective functions are defined. As there are no data sets of benchmark instances available for the solution of this kind of multiobjective VRP problems, we created a number of instances as follows. Initially, we took five instances with the coordinates of 100 cities from the TSPLIB (kroA100, kroB100, kroC100, kroD100, and kroE100). The data for the capacity, the time limits and the demands were taken from the third instance (par3) of the classic Christofides benchmark instances [11] used for the solution of the capacitated vehicle routing problem (CVRP).

Thus, we created a new set of instances combining the Kro#100 instances (where # corresponds to A or B or C or D or E) with the par3 instance where the coordinates of 100 nodes are taken from the corresponding kro#100 data set and the corresponding demand of each of the 100 nodes, the maximum tour length, the service time and the capacity of each vehicle were taken from the par3 instance. Also, the maximum tour length, the service time and the capacity of each vehicle were taken from the par3 instance. The combination of these five instances are used for the creation of the instances for the 2-objective functions problems. For example, in order to create a 2-objective functions asymmetric problem instance, we use for the calculation of the values of the first objective function data from the combination kroA100par3, while for the calculation of the values of the second objective function we use data from the combinations kroB100par3, kroC100par3 and kroD100par3. For the production of the route parameters (r_{ij}) Table of the second objective function the necessary data are taken from kroC100par3, kroD100par3 and kroB100par3 as follows. The route

parameters of the network is an asymmetric table with positive numbers. In order to create those numbers we divide the route parameters' Table in two parts. In the first part all elements that are in the down side from the main diagonal are calculated using the Euclidean distance between the nodes corresponding to the kroC100par3 set and in the second part all elements that are at the top of the main diagonal are calculated using the Euclidean distance between the nodes corresponding to the kroD100par3 set. Then, each element of the route parameters' Table is divided with the corresponding element of a table created by calculating the Euclidean distance between the nodes corresponding to the kroB100par3 set.

In order to create a route for every objective function the time and the capacity constraints are needed. In case of solving a 2-objective functions symmetric problem with first objective function the OF1 and second objective function the OF2 (OF1-OF2 for the MSDRFCVRP) or the OF3 (OF1-OF3 for the MSPRFCVRP), the combination "A-B" in all Tables and Figures means that A corresponds to KroA100par3 and B corresponds to kroB100par3. Generally in a problem when the OF1 objective function is used the Euclidean distances of the corresponding nodes of the instance correspond to the time durations. In the case that a 2-objective functions asymmetric problem is selected to be solved with first objective function the OF1 and second the OF2 (OF1-OF2 for the MADRFCVRP) or the OF3 (OF1-OF3 for the MAPRFCVRP) the notation for each new generated instance in all corresponding tables includes a combination of 4 different instances. Thus, a notation "A-B-CD", means that for the calculation of the time duration the data from kroA100par3 are used, for the calculation of the distance the data from kroB100par3 were used while for the calculation of the route parameters needed for the OF2 (or OF3) objective function, the data from kroC100par3, kroD100par3 and kroB100par3 were used as it was described previously.

The parameters of all the algorithms were selected after testing with different values and the ones selected are those that gave the best computational results, taking into account the quality of the solutions and the computational time needed to achieve these solutions and, also, taking into account the fact that we would like to test the algorithms with the same function evaluations. Thus, the selected parameters for each algorithm are given in the following:

Parallel multi-start NSDEs

- Number of individuals for each initial population: 100.
- Number of generations: 500.
- Number of initial populations: 10.
- $\beta = 0.5$

Parallel multi-start NSGA II

- Number of individuals for each initial population: 100.
- Number of generations: 500.
- Number of initial populations: 10.

After the selection of the final parameters, the four algorithms were tested for ten combinations for the two objective functions. In the following tables the comparisons performed based on the four evaluation measures presented previously are presented

Table 2 Results of the first three measures for the four algorithms in ten instances when the symmetric delivery problem using objective functions OF1-OF2 is solved

Multiobjective symmetric delivery route based fuel consumption VRP												
OF1-OF2	PMS-NSDE1			PMS-NSDE2			PMS-NSDE3			PMS-NSGA II		
	<i>L</i>	<i>M_k</i>	Δ	<i>L</i>	<i>M_k</i>	Δ	<i>L</i>	<i>M_k</i>	Δ	<i>L</i>	<i>M_k</i>	Δ
A-B	<u>55</u>	<u>583.11</u>	<u>0.70</u>	48	575.55	0.72	51	575.55	0.72	58	585.38	0.66
A-C	43	590.45	0.69	<u>47</u>	587.11	0.68	<u>47</u>	605.90	0.62	55	587.85	0.65
A-D	<u>49</u>	577.26	0.587	48	568.21	0.61	42	582.62	0.68	57	572.40	0.588
A-E	36	572.06	0.57	33	579.87	0.70	52	590.56	0.69	47	586.90	0.61
B-C	50	586.03	0.58	44	572.77	0.61	<u>52</u>	579.81	0.70	68	573.00	0.61
B-D	39	587.80	<u>0.69</u>	<u>46</u>	567.84	0.75	40	581.86	0.70	61	571.94	0.56
B-E	48	584.47	0.66	59	596.16	0.76	50	595.06	<u>0.63</u>	50	554.21	0.51
C-D	52	582.73	0.66	59	573.96	0.75	52	571.73	0.72	50	578.72	0.71
C-E	<u>51</u>	596.56	0.52	43	572.21	0.78	40	588.11	0.78	55	558.80	0.68
D-E	63	580.33	0.66	49	591.55	0.50	48	604.42	0.74	54	601.82	0.59

and in the figures a selected number of Pareto fronts are given. More precisely, we use for evaluation measures the number of solutions (*L*) in the non-dominated set, the maximum extend in each dimension (*M_k*), the minimization of the spread of solutions (Δ) and the Coverage (*C*). In all the tables except those that contain the values of the Coverage the best value for each measure from the comparison of all algorithms is signed as bold while from the comparison of the three PMS-NSDE algorithms the best values are underlined in the tables. In general, it is preferred the *L*, the *M_k* and the *C* measures to be as larger as possible and the Δ value to be as smaller as possible.

In Tables 2, 3, 4 and 5, the results of the four measures for the four algorithms in ten combinations of instances for the multiobjective delivery route-based fuel consumption problems are presented. More precisely, in Table 2 and in Table 4 the results of the symmetric problem (OF1-OF2 for the MSDRFCVRP) (Table 2) and of the asymmetric problem (OF1-OF2 for the MADRFCVRP) (Table 4) using the first two objective functions for the first three evaluation measures are presented, respectively. On the other hand, in Tables 3 and 5 the results for the last evaluation measure (*C*) for the same formulations (symmetric and asymmetric), same objective functions and same instances as in Tables 2 and 4 are presented, respectively. In Tables 6, 7, 8 and 9 the results of the four measures for the four algorithms for ten combinations of instances for the multiobjective pick-up route-based fuel consumption problem are presented. More precisely, in Tables 6 and 8 the results of the first three measures in the symmetric (Table 6) and in the asymmetric (Table 8) case are given respectively while in Tables 7 and 9 the results of the fourth measure in the symmetric (Table 7) and in the asymmetric (Table 9) case are presented, respectively. In Fig. 1, the Pareto Fronts of the symmetric delivery problem using objective functions OF1-OF2 (first part of the figure) and of the symmetric pick-up problem using objective functions OF1-OF3 (second part of the figure) for the instance “D-E” for all algo-

Table 3 Results of the *C* measure for the four algorithms in ten instances when the symmetric delivery problem using objective functions OF1-OF2 is solved

OF1-OF2	Multiobjective symmetric delivery route based fuel consumption VRP								
A-B	NSDE1	NSDE2	NSDE3	NSGA II	B-D	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.271	0.255	0.966	NSDE1	0	0.522	0.450	0.934
NSDE2	0.436	0	0.353	0.931	NSDE2	0.410	0	0.425	0.885
NSDE3	0.618	0.396	0	0.931	NSDE3	0.308	0.261	0	0.820
NSGA II	0.018	0.042	0	0	NSGA II	0	0.043	0.050	0
A-C	NSDE1	NSDE2	NSDE3	NSGA II	B-E	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.255	0.277	0.836	NSDE1	0	0.390	0.500	0.720
NSDE2	0.605	0	0.468	0.855	NSDE2	0.521	0	0.260	0.740
NSDE3	0.488	0.362	0	0.855	NSDE3	0.438	0.508	0	0.820
NSGA II	0.047	0.021	0.064	0	NSGA II	0.250	0.220	0.100	0
A-D	NSDE1	NSDE2	NSDE3	NSGA II	C-D	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.208	0.429	0.877	NSDE1	0	0.542	0.731	0.940
NSDE2	0.367	0	0.500	0.895	NSDE2	0.346	0	0.462	0.960
NSDE3	0.531	0.417	0	0.842	NSDE3	0.231	0.492	0	0.920
NSGA II	0.122	0.104	0.095	0	NSGA II	0.038	0	0.058	0
A-E	NSDE1	NSDE2	NSDE3	NSGA II	C-E	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.394	0.462	0.936	NSDE1	0	0.465	0.275	0.800
NSDE2	0.528	0	0.385	1.000	NSDE2	0.549	0	0.475	0.927
NSDE3	0.500	0.455	0	0.979	NSDE3	0.549	0.488	0	0.818
NSGA II	0.028	0	0	0	NSGA II	0.235	0	0.075	0
B-C	NSDE1	NSDE2	NSDE3	NSGA II	D-E	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.523	0.596	0.853	NSDE1	0	0.367	0.292	0.852
NSDE2	0.320	0	0.462	0.809	NSDE2	0.492	0	0.250	0.870
NSDE3	0.240	0.273	0	0.824	NSDE3	0.413	0.429	0	0.926
NSGA II	0.140	0.068	0.019	0	NSGA II	0.032	0.041	0.042	0

rithms are presented, respectively while in Fig. 2 the Pareto Fronts of the asymmetric delivery problem using objective functions OF1-OF2 (first part of the figure) and of the asymmetric pick-up problem using objective functions OF1-OF3 (second part of the figure) for the instance “D-E-BC” for all algorithms are presented, respectively. In Fig. 3, five runs of the algorithms PMS-NSDE1 and PMS-NSGA II for the instance “A-B” of the MSDRFCVRP are presented in order to prove the stability of the algorithms. In all the Tables that present results for the *C* measure the notation of the algorithms PMS-NSDE# and PMS-NSGA II have been replaced with the notations NSDE# and NSGA II, respectively, in order to reduce the size of the Tables.

In Table 2, the results of the four algorithms solving the first two objective functions symmetric delivery problem (OF1-OF2 for the MSDRFCVRP) are presented. The PMS-NSGA II performs better than the other algorithms for the *L* measure (it performs better in six out of ten instances). The PMS-NSDE1 performs better for the Δ measure

Table 4 Results of the first three measures for the four algorithms in ten instances when the asymmetric delivery problem using objective functions OF1-OF2 is solved

Multiobjective asymmetric delivery route based fuel consumption VRP												
OF1-OF2	PMS-NSDE1			PMS-NSDE2			PMS-NSDE3			PMS-NSGA II		
	<i>L</i>	<i>M_k</i>	Δ	<i>L</i>	<i>M_k</i>	Δ	<i>L</i>	<i>M_k</i>	Δ	<i>L</i>	<i>M_k</i>	Δ
A-B-CD	<u>46</u>	577.99	0.65	37	582.07	0.70	36	596.81	0.54	54	578.84	0.60
A-C-BD	45	604.90	0.71	<u>50</u>	609.60	0.49	<u>50</u>	602.51	0.70	56	591.93	0.60
A-D-BE	48	586.83	0.79	44	579.71	0.62	39	612.26	0.65	43	603.98	0.63
A-E-BD	44	585.43	0.71	<u>51</u>	592.32	0.63	49	602.16	0.70	58	594.59	0.70
B-C-AD	<u>46</u>	586.36	0.64	44	596.16	0.63	42	601.53	0.60	57	595.61	0.68
B-D-AC	54	589.29	0.616	49	574.38	0.63	63	<u>592.42</u>	0.64	55	608.36	0.623
B-E-AD	44	572.54	0.60	<u>51</u>	584.35	0.65	40	584.13	0.50	57	560.85	0.56
C-D-AE	<u>51</u>	570.63	0.66	48	584.78	0.54	45	590.64	0.68	52	569.04	0.64
C-E-AB	40	584.51	0.69	40	589.07	0.59	<u>45</u>	574.86	<u>0.58</u>	49	572.14	0.51
D-E-BC	<u>43</u>	584.14	0.72	37	597.06	0.55	<u>43</u>	580.91	0.54	51	568.55	0.59

(it performs better in five instances) and it has equal performance with the PMS-NSDE3 for the M_k measure. If we would like to compare the results of the three versions of the PMS-NSDE algorithms, the PMS-NSDE1 performs slightly better than the other two algorithms according to the M_k and Δ measures and has equal performance with the PMS-NSDE2 for the L measure. According to the coverage table (Table 3) the PMS-NSDE3's Pareto Fronts dominate the Pareto Fronts produced from the other three algorithms.

When the asymmetric delivery problem (OF1-OF2 for the MADRFCVRP) is solved (Tables 4, 5) the results are slightly different from the previous case. The PMS-NSGA II performs better than the other three algorithms only for the L measure and taking into account the M_k and the Δ measures the PMS-NSDE3 performs better than the other algorithms in five and four instances, respectively. In the comparison of PMS-NSDE variants, according to the number of Pareto solutions L the PMS-NSDE1 performs better in five instances and the PMS-NSDE3 performs better in six instances considering the M_k measure and in five instances considering the Δ measure. According to the Coverage Table (Table 5), the results are different from the previous symmetric problem, thus, the PMS-NSDE2's Pareto Fronts dominates the Pareto Fronts produced from the other three algorithms.

A very interesting observation is derived when the results of the symmetric pick-up problem (using objective functions OF1-OF3 for the MSPRFCVRP) are presented (Tables 6, 7). The PMS-NSGA II performs better than the other three algorithms for the number of the Pareto front solutions L (better performance in 9 instances) and the PMS-NSDE3 performs better for the M_k measure (better performance in 6 instances), that is almost the same outcome as in the case when the symmetric and the asymmetric delivery problems are solved. Also, PMS-NSGA II performs better for the Δ measure (better performance in 7 instances). From the comparison of the three variants of the PMS-NSDE we conclude that according to the number of Pareto

Table 5 Results of the C measure for the four algorithms in ten instances when the asymmetric delivery problem using objective functions OF1-OF2 is solved

OF1-OF2	Multiobjective asymmetric delivery route based fuel consumption VRP								
A-B-CD	NSDE1	NSDE2	NSDE3	NSGA II	B-D-AC	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.162	0.222	0.630	NSDE1	0	0.531	0.460	0.818
NSDE2	0.609	0	0.444	0.759	NSDE2	0.296	0	0.254	0.764
NSDE3	0.652	0.486	0	0.796	NSDE3	0.370	0.510	0	0.727
NSGA II	0.065	0.135	0.111	0	NSGA II	0.056	0.102	0.159	0
A-C-BD	NSDE1	NSDE2	NSDE3	NSGA II	B-E-AD	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.380	0.600	0.786	NSDE1	0	0.569	0.325	0.772
NSDE2	0.533	0	0.600	0.768	NSDE2	0.273	0	0.275	0.719
NSDE3	0.400	0.280	0	0.679	NSDE3	0.523	0.588	0	0.807
NSGA II	0.067	0.160	0.120	0	NSGA II	0.091	0.098	0.175	0
A-D-BE	NSDE1	NSDE2	NSDE3	NSGA II	C-D-AE	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.273	0.231	0.907	NSDE1	0	0.292	0.400	0.923
NSDE2	0.417	0	0.538	0.977	NSDE2	0.392	0	0.356	0.827
NSDE3	0.500	0.318	0	0.907	NSDE3	0.471	0.479	0	0.846
NSGA II	0.042	0	0.128	0	NSGA II	0.020	0.021	0.089	0
A-E-BD	NSDE1	NSDE2	NSDE3	NSGA II	C-E-AB	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.275	0.429	0.655	NSDE1	0	0.275	0.378	0.653
NSDE2	0.659	0	0.633	0.707	NSDE2	0.350	0	0.311	0.898
NSDE3	0.295	0.216	0	0.569	NSDE3	0.350	0.500	0	0.837
NSGA II	0.182	0.118	0.122	0	NSGA II	0.100	0.075	0.089	0
B-C-AD	NSDE1	NSDE2	NSDE3	NSGA II	D-E-BC	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.500	0.524	0.860	NSDE1	0	0.270	0.209	0.745
NSDE2	0.348	0	0.310	0.895	NSDE2	0.558	0	0.395	0.863
NSDE3	0.435	0.455	0	0.825	NSDE3	0.581	0.378	0	0.843
NSGA II	0.022	0.023	0	0	NSGA II	0.070	0.027	0.070	0

front solutions L the PMS-NSDE2 has equal performance with the PMS-NSDE3 and they perform better than the PMS-NSDE1 in four instances, respectively. The PMS-NSDE3 performs better for the M_k and Δ measures than the other two algorithms in seven instances respectively. This outcome is very important as we can see that the performance of the PMS-NSDE3 algorithm in these measures is equally well whether we solve an asymmetric delivery or a symmetric pick-up problem. Also According to the coverage table (Table 7), the results are different from the previous symmetric problem, thus, the PMS-NSDE1's and the PMS-NSDE2's Pareto Fronts dominate the Pareto Fronts produced from the other two algorithms in four instances, respectively.

Finally, when the asymmetric pick-up problem (objective functions OF1-OF3 for the MAPRFCVRP) is solved (Tables 8, 9) the results lead us to the following conclusions. The PMS-NSGA II performs better than the other three algorithms in the number of the Pareto front solutions L (better performance in 7 instances) and for the

Table 6 Results of the first three measures for the four algorithms in ten instances when the symmetric pick-up problem using objective functions OF1-OF3 is solved

Multiobjective symmetric pick-up route based fuel consumption VRP												
OF1-OF3	PMS-NSDE1			PMS-NSDE2			PMS-NSDE3			PMS-NSGA II		
	<i>L</i>	<i>M_k</i>	Δ	<i>L</i>	<i>M_k</i>	Δ	<i>L</i>	<i>M_k</i>	Δ	<i>L</i>	<i>M_k</i>	Δ
A-B	<u>52</u>	601.07	0.74	43	593.49	0.73	46	608.52	0.63	55	588.31	0.68
A-C	45	589.05	0.70	37	595.05	0.75	46	601.55	<u>0.67</u>	60	586.84	0.62
A-D	46	592.16	0.70	40	590.56	0.68	48	596.19	<u>0.64</u>	60	578.07	0.58
A-E	41	575.09	0.68	44	591.40	<u>0.64</u>	49	<u>598.45</u>	0.72	53	603.91	0.57
B-C	46	581.79	0.76	<u>52</u>	567.17	<u>0.63</u>	47	589.30	0.72	63	581.16	0.55
B-D	47	572.06	0.70	54	<u>583.13</u>	0.70	50	562.45	<u>0.641</u>	51	593.40	0.636
B-E	51	589.09	0.68	<u>53</u>	592.54	0.66	39	604.38	0.62	59	592.66	0.66
C-D	40	583.02	0.71	<u>41</u>	585.14	0.70	40	589.16	<u>0.62</u>	50	561.90	0.61
C-E	45	<u>597.67</u>	0.569	41	589.82	0.568	<u>47</u>	590.05	0.71	48	606.36	0.60
D-E	<u>48</u>	<u>605.37</u>	0.69	45	588.50	0.69	47	604.34	<u>0.65</u>	76	619.66	0.58

Δ measure (better performance in 5 instances) that is almost the same outcome as in the case when the symmetric pick-up problem is solved. The PMS-NSDE3 performs better for the M_k measure (better performance in 5 instances) and the Δ measure (better performance in 4 instances). This outcome is very important as we can see that the performance of the PMS-NSDE3 algorithm in the M_k measure is equally well in every problem and in the Δ measure is equally well whether we solve an asymmetric delivery or an asymmetric pick-up problem is solved. In the comparison of PMS-NSDE variants, the outcome is almost the same as in the case when the asymmetric delivery problem is solved. According to the number of Pareto solutions L the PMS-NSDE1 performs better in six instances and the PMS-NSDE3 performs better in five instances considering the M_k measure and in six instances considering the Δ measure. According to the Coverage Table (Table 9) all the PMS-NSDE's Pareto Fronts have equal performance.

In general, based on all Tables the PMS-NSGA II algorithm produces Pareto front with larger numbers of solutions and better distribution than the other algorithms. The PMS-NSDE3 algorithm produces more extended Pareto fronts and the Pareto fronts produced by PMS-NSDE2 dominate the fronts produced from the other algorithms. From the comparison of the three PMS-NSDE algorithms we conclude that considering the L measure the PMS-NSDE1 performs slightly better than the other two algorithms as it performs better in 42.5 % of the instances while the PMS-NSDE3 and the PMS-NSDE2 perform better in 35 and 32.5 %, respectively. Considering the M_k measure the PMS-NSDE3 performs slightly better than the others as it performs better in 55 % of the instances while the PMS-NSDE1 and the PMS-NSDE2 perform better in 25 and 20 %, respectively. Taking into account the Δ measure the PMS-NSDE3 performs slightly better than the others as it performs better in 50 % of the instances while the PMS-NSDE1 and the PMS-NSDE2 perform better in 30 and 20 %, respectively. Finally, considering the C measure the PMS-NSDE2 performs slightly better than the other

Table 7 Results of the C measure for the four algorithms in ten instances when the symmetric pick-up problem using objective functions OF1-OF3 is solved

OF1-OF3	Multiobjective Symmetric Pick-up Route based Fuel Consumption VRP								
A-B	NSDE1	NSDE2	NSDE3	NSGA II	B-D	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.442	0.500	0.836	NSDE1	0	0.519	0.580	0.902
NSDE2	0.500	0	0.478	0.945	NSDE2	0.447	0	0.480	0.843
NSDE3	0.346	0.372	0	0.873	NSDE3	0.383	0.315	0	0.804
NSGA II	0.058	0.047	0.087	0	NSGA II	0	0	0.020	0
A-C	NSDE1	NSDE2	NSDE3	NSGA II	B-E	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.459	0.500	0.917	NSDE1	0	0.358	0.308	0.814
NSDE2	0.444	0	0.500	0.900	NSDE2	0.549	0	0.590	0.864
NSDE3	0.267	0.297	0	0.867	NSDE3	0.529	0.358	0	0.831
NSGA II	0	0.054	0.109	0	NSGA II	0.059	0.038	0.103	0
A-D	NSDE1	NSDE2	NSDE3	NSGA II	C-D	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.350	0.375	0.833	NSDE1	0	0.610	0.550	0.900
NSDE2	0.370	0	0.396	0.833	NSDE2	0.125	0	0.300	0.880
NSDE3	0.543	0.475	0	0.933	NSDE3	0.175	0.512	0	0.820
NSGA II	0.022	0.025	0.104	0	NSGA II	0.025	0.073	0.025	0
A-E	NSDE1	NSDE2	NSDE3	NSGA II	C-E	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.568	0.551	0.962	NSDE1	0	0.366	0.426	0.708
NSDE2	0.341	0	0.429	0.887	NSDE2	0.689	0	0.638	0.813
NSDE3	0.341	0.386	0	0.943	NSDE3	0.511	0.171	0	0.667
NSGA II	0	0	0	0	NSGA II	0.156	0.073	0.149	0
B-C	NSDE1	NSDE2	NSDE3	NSGA II	D-E	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.462	0.489	0.952	NSDE1	0	0.289	0.404	0.908
NSDE2	0.457	0	0.319	1.000	NSDE2	0.583	0	0.617	0.921
NSDE3	0.543	0.442	0	0.905	NSDE3	0.375	0.267	0	0.908
NSGA II	0	0	0.021	0	NSGA II	0.021	0	0	0

two algorithms as it performs better in 40 % of the instances while the PMS-NSDE1 and PMS-NSDE2 follow by performing better in 37.5 and 37.5 % of the instances, respectively. For the L and C measures the summation of the percentages is larger than 1 considering that in some instances two or more algorithms perform equal better.

A statistical analysis based on the Friedman test [26,27] for all algorithms is presented in Table 10 based on the results of Tables 2, 4, 6, 8. In each row of the Table the rank of the algorithms based on the values of Tables 2, 4, 6, 8 in each instance is presented. In the last row the average ranks for all algorithms are given. In the Friedman test all the algorithms are ranked for each data set separately [18,28] as it is shown in Table 10. Let r_i^j be the rank of the j th of k algorithms on the i th of N data sets, where in our study k is equal to 4 and N is equal to 40 (if we analyze each of the three metrics separately) and 120 (if we analyze the three metrics together). The Friedman test compares the average ranks of algorithms, $R_j = \frac{1}{N} \sum_i r_i^j$. Under the null-hypothesis [18,28], which states that all the algorithms are equivalent and, thus, their ranks R_j should be equal, the Friedman statistic

Table 8 Results of the first three measures for the four algorithms in ten instances when the asymmetric pick-up problem using objective functions OF1-OF3 is solved

Multiobjective asymmetric pick-up route based fuel consumption VRP												
OF1-OF3	PMS-NSDE1			PMS-NSDE2			PMS-NSDE3			PMS-NSGA II		
	<i>L</i>	<i>M_k</i>	Δ	<i>L</i>	<i>M_k</i>	Δ	<i>L</i>	<i>M_k</i>	Δ	<i>L</i>	<i>M_k</i>	Δ
A-B-CD	58	587.41	<u>0.66</u>	51	583.50	0.73	55	587.94	0.68	46	585.60	0.55
A-C-BD	<u>50</u>	611.46	0.73	49	565.36	0.70	<u>50</u>	611.52	<u>0.59</u>	54	601.11	0.51
A-D-BE	<u>47</u>	<u>605.58</u>	0.68	38	595.68	0.69	44	604.93	<u>0.66</u>	55	615.67	0.64
A-E-BD	56	590.84	0.58	48	597.71	0.67	47	606.71	0.66	53	590.10	0.74
B-C-AD	47	584.72	0.63	57	580.18	0.62	56	593.39	0.60	42	580.71	0.64
B-D-AC	<u>42</u>	575.56	0.56	39	585.05	0.60	41	576.58	0.46	59	579.43	0.61
B-E-AD	36	569.59	0.58	46	588.14	0.60	55	603.08	0.73	55	586.54	0.67
C-D-AE	61	618.99	<u>0.58</u>	48	603.31	0.69	34	617.29	0.64	55	617.19	0.56
C-E-AB	48	594.14	0.66	49	593.86	0.71	34	587.99	0.53	48	572.56	0.58
D-E-BC	51	577.45	0.77	48	578.90	0.70	<u>53</u>	573.07	0.59	55	568.12	0.65

$$\chi_F^2 = \frac{12N}{k(k+1)} \left[\sum_j R_j^2 - \frac{k(k+1)^2}{4} \right] \tag{19}$$

is distributed according to χ_F^2 with $k - 1$ degrees of freedom, when N and k are big enough. Iman and Davenport [31] showed that Friedmans χ_F^2 is undesirably conservative [18] and derived a better statistic [31]

$$F_F = \frac{(N - 1)\chi_F^2}{N(k - 1) - \chi_F^2} \tag{20}$$

which is distributed according to the F -distribution with $k - 1$ and $(k - 1) \times (N - 1)$ degrees of freedom [18]. In our statistical analysis, both equations were used and the values found were: $\chi_F^2 = 31.825$ and $F_F = 14.074$ for the L metric, $\chi_F^2 = 14.272$ and $F_F = 5.264$ for the M_k metric and $\chi_F^2 = 12.877$ and $F_F = 4.688$ for the Δ metric, while when the results of all metrics are analyzed together we found: $\chi_F^2 = 24.93$ and $F_F = 8.853$ respectively. From the statistical Tables and for $\alpha = 0.05$ the critical values are for χ^2 with $k - 1=3$ degrees of freedom equal to 0.35. The $F(k - 1, (k - 1) \times (N - 1)) = F(3, 40)$ is equal to 2.84 for each metric separately and when all the metrics are analyzed together the $F(3, 120)$ is equal to 2.68. Thus, we reject the null hypothesis in any case. From this analysis, we can say that there are statistical differences between the algorithms.

6 Which solution of the Pareto front to choose?

In real world problems after the production of the Pareto front solutions the decision maker has to make the next step and to choose one solution from these solutions

Table 9 Results of the *C* measure for the four algorithms in ten instances when the asymmetric pick-up problem using objective functions OF1-OF3 is solved

OF1-OF3	Multiobjective asymmetric pick-up route based fuel consumption VRP								
A-B-CD	NSDE1	NSDE2	NSDE3	NSGA II	B-D-AC	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.588	0.545	0.913	NSDE1	0	0.590	0.220	0.881
NSDE2	0.414	0	0.600	0.761	NSDE2	0.238	0	0.146	0.729
NSDE3	0.310	0.275	0	0.696	NSDE3	0.429	0.692	0	0.915
NSGA II	0.052	0.078	0.127	0	NSGA II	0.143	0.077	0.024	0
A-C-BD	NSDE1	NSDE2	NSDE3	NSGA II	B-E-AD	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.306	0.440	0.833	NSDE1	0	0.478	0.618	0.945
NSDE2	0.600	0	0.520	0.796	NSDE2	0.333	0	0.527	0.873
NSDE3	0.320	0.245	0	0.778	NSDE3	0.083	0.413	0	0.891
NSGA II	0.060	0.102	0.040	0	NSGA II	0.028	0.043	0.036	0
A-D-BE	NSDE1	NSDE2	NSDE3	NSGA II	C-D-AE	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.395	0.432	0.873	NSDE1	0	0.271	0.029	0.764
NSDE2	0.362	0	0.636	0.818	NSDE2	0.672	0	0.176	0.836
NSDE3	0.489	0.368	0	0.782	NSDE3	0.820	0.563	0	0.927
NSGA II	0.021	0.026	0.068	0	NSGA II	0.066	0.042	0.059	0
A-E-BD	NSDE1	NSDE2	NSDE3	NSGA II	C-E-AB	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.417	0.638	0.811	NSDE1	0	0.184	0.118	0.771
NSDE2	0.357	0	0.660	0.792	NSDE2	0.583	0	0.294	0.729
NSDE3	0.268	0.292	0	0.792	NSDE3	0.625	0.571	0	0.708
NSGA II	0.250	0.146	0.106	0	NSGA II	0.104	0.163	0.147	0
B-C-AD	NSDE1	NSDE2	NSDE3	NSGA II	D-E-BC	NSDE1	NSDE2	NSDE3	NSGA II
NSDE1	0	0.316	0.554	0.667	NSDE1	0	0.229	0.321	0.691
NSDE2	0.574	0	0.625	0.595	NSDE2	0.725	0	0.491	0.727
NSDE3	0.383	0.316	0	0.500	NSDE3	0.588	0.479	0	0.800
NSGA II	0.213	0.228	0.393	0	NSGA II	0.118	0.167	0.208	0

considering his preferences. For example, suppose that the decision maker is the president of a company that needs a fast and efficient method to help him/her to select the most suitable solution of a Pareto front. Using the following proposed method the most suitable solution, considering the preferences of the decision maker, could be selected.

Initially, the decision maker is asked to evaluate a number of *K Utility Variables (UV)*. The number of the *Utility Variables* is equal to the number of the objective functions of the problem. Each one of those variables corresponds to an objective function or criterion and it can take any value between [0,1]. The values of these variables represent how useful for the decision maker is the best value of each criterion. The closer the value of the *Utility Variable* to zero, the less useful is the criterion for the decision maker. On the other hand, the closer the value of the *Utility Variable* to one, the more useful is the criterion for the decision maker. Considering that we conclude to the following results solving a multiobjective problem:

Solutions	Pareto Front Time (min)	Fuel (lt)
1	100	2
2	70	3
3	50	5
4	30	7
5	10	10

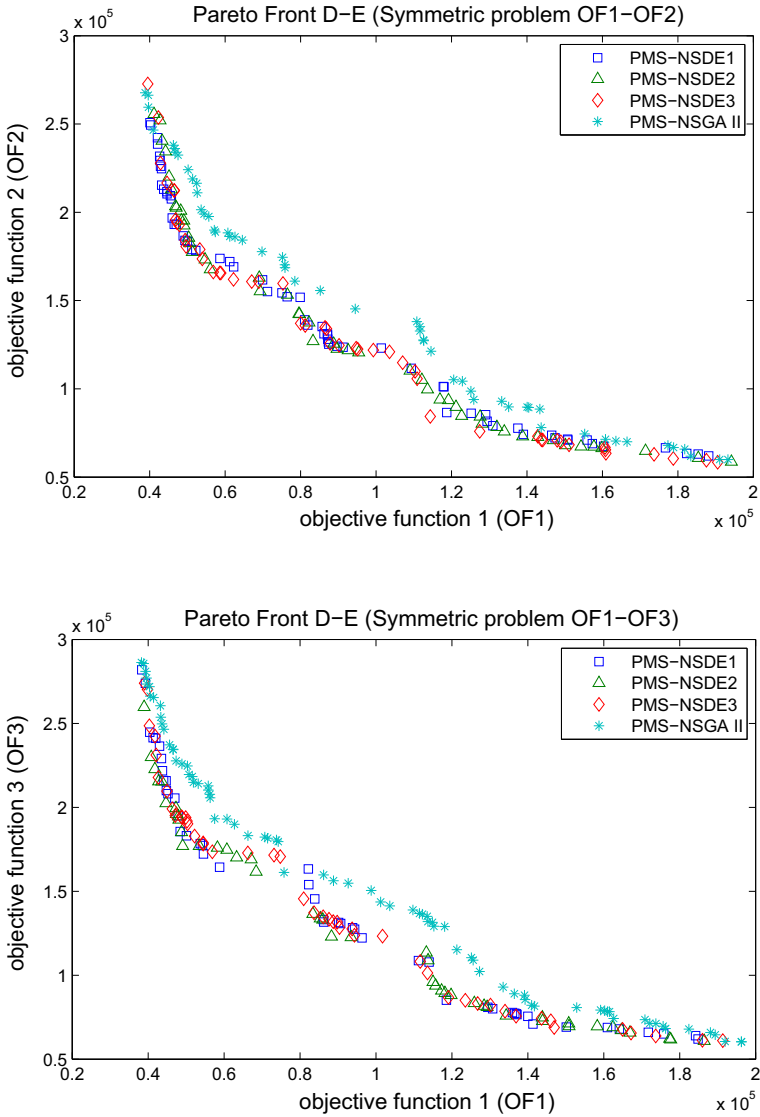


Fig. 1 Pareto fronts of the four algorithms for the instance “D-E”

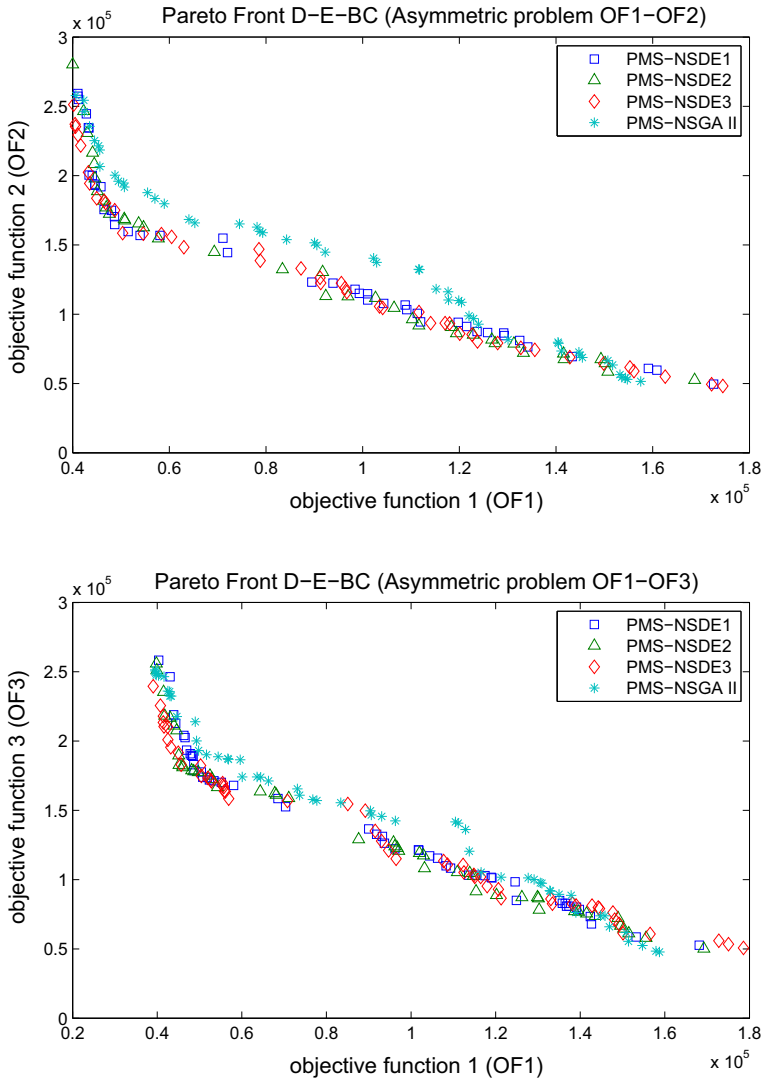


Fig. 2 Pareto fronts of the four algorithms for the instance “D-E-BC”

Considering that, for the decision maker it is very important to select a solution that results to a low fuel consumption route and, also, that for the decision maker is not so important the time needed to complete the route. Thus, the *Utility Variables* could be $UV(Fuel) = 0.9$ for the Fuel criterion and $UV(Time) = 0.1$ for the Time criterion. At the next step, we use the following equations in order to find each one of the coordinates of the *Utility Point (UP)* on the Pareto Front diagram (Fig. 4):

$$UP(Time) = (\text{Max Time value} - \text{Min Time value}) * (1 - UV(Time)) + \text{Min Time value} = (100 - 10) * 0.9 + 10 = 91$$

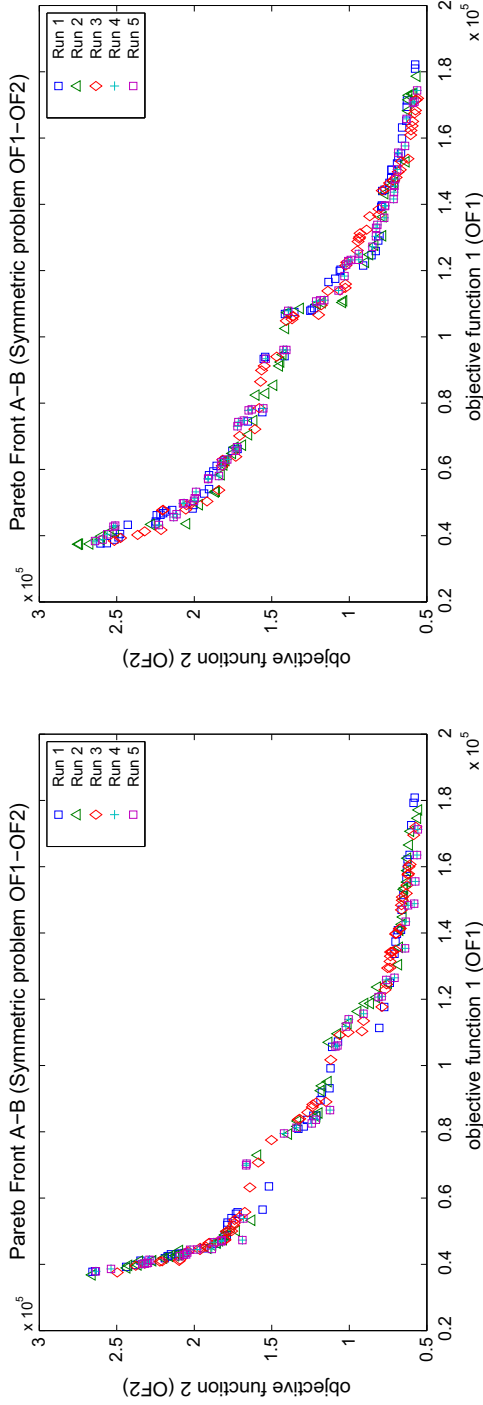


Fig. 3 Pareto fronts for the instance A-B when the symmetric delivery problem using objective functions OF1-OF2 is solved in five different runs of PMS-NSDE1 and PMS-NSGA II algorithms

Table 10 Results of Friedman test for all algorithm

	PMS-NSDE1			PMS-NSDE2			PMS-NSDE3			PMS-NSGA II		
	L	M_k	Δ	L	M_k	Δ	L	M_k	Δ	L	M_k	Δ
OF1-OF2	Multiobjective Symmetric Delivery Route-based Fuel Consumption Vehicle Routing Problem											
A-B	2.00	2.00	2.00	4.00	3.50	3.50	3.00	3.50	3.50	1.00	1.00	1.00
A-C	4.00	2.00	4.00	2.50	4.00	3.00	2.50	1.00	1.00	1.00	3.00	2.00
A-D	2.00	2.00	1.00	3.00	4.00	3.00	4.00	1.00	4.00	1.00	3.00	2.00
A-E	3.00	4.00	1.00	4.00	3.00	4.00	1.00	1.00	3.00	2.00	2.00	2.00
B-C	3.00	1.00	1.00	4.00	4.00	2.50	2.00	2.00	4.00	1.00	3.00	2.50
B-D	4.00	1.00	2.00	2.00	4.00	4.00	3.00	2.00	3.00	1.00	3.00	1.00
B-E	4.00	3.00	3.00	1.00	1.00	4.00	2.50	2.00	2.00	2.50	4.00	1.00
C-D	2.50	1.00	1.00	1.00	3.00	4.00	2.50	4.00	3.00	4.00	2.00	2.00
C-E	2.00	1.00	1.00	3.00	3.00	3.50	4.00	2.00	3.50	1.00	4.00	2.00
D-E	1.00	4.00	3.00	3.00	3.00	1.00	4.00	1.00	4.00	2.00	2.00	2.00
OF1-OF2	Multiobjective asymmetric delivery route-based fuel consumption vehicle routing problem											
A-B-CD	2.00	4.00	3.00	3.00	2.00	4.00	4.00	1.00	1.00	1.00	3.00	2.00
A-C-BD	4.00	2.00	4.00	2.50	1.00	1.00	2.50	3.00	3.00	1.00	4.00	2.00
A-D-BE	1.00	3.00	4.00	2.00	4.00	1.00	4.00	1.00	3.00	3.00	2.00	2.00
A-E-BD	4.00	4.00	4.00	2.00	3.00	1.00	3.00	1.00	2.50	1.00	2.00	2.50
B-C-AD	2.00	4.00	3.00	3.00	2.00	2.00	4.00	1.00	1.00	1.00	3.00	4.00
B-D-AC	3.00	3.00	1.00	4.00	4.00	3.00	1.00	2.00	4.00	2.00	1.00	2.00
B-E-AD	3.00	3.00	3.00	2.00	1.00	4.00	4.00	2.00	1.00	1.00	4.00	2.00
C-D-AE	2.00	3.00	3.00	3.00	2.00	1.00	4.00	1.00	4.00	1.00	4.00	2.00
C-E-AB	3.50	2.00	4.00	3.50	1.00	3.00	2.00	3.00	2.00	1.00	4.00	1.00
D-E-BC	2.50	2.00	4.00	4.00	1.00	2.00	2.50	3.00	1.00	1.00	4.00	3.00

Table 10 continued

	PMS-NSDE1			PMS-NSDE2			PMS-NSDE3			PMS-NSGA II		
	L	M _k	Δ	L	M _k	Δ	L	M _k	Δ	L	M _k	Δ
OF1-OF3	Multiobjective symmetric pick-up route-based fuel consumption vehicle routing problem											
A-B	2.00	2.00	4.00	4.00	4.00	3.00	3.00	1.00	1.00	1.00	3.00	2.00
A-C	3.00	3.00	3.00	4.00	2.00	4.00	2.00	1.00	1.00	1.00	4.00	1.00
A-D	3.00	2.00	4.00	4.00	3.00	3.00	2.00	1.00	1.00	1.00	4.00	1.00
A-E	4.00	4.00	3.00	3.00	3.00	2.00	2.00	2.00	2.00	1.00	1.00	1.00
B-C	4.00	2.00	4.00	2.00	4.00	2.00	3.00	1.00	1.00	1.00	3.00	1.00
B-D	4.00	3.00	3.50	1.00	2.00	3.50	3.00	3.00	2.00	2.00	1.00	1.00
B-E	3.00	4.00	4.00	2.00	3.00	2.50	4.00	1.00	1.00	1.00	2.00	2.50
C-D	3.50	3.00	4.00	2.00	2.00	3.00	3.50	1.00	2.00	1.00	4.00	1.00
C-E	3.00	2.00	2.00	4.00	4.00	1.00	2.00	3.00	4.00	1.00	1.00	3.00
D-E	2.00	2.00	3.50	4.00	4.00	3.50	3.00	3.00	2.00	1.00	1.00	1.00
OF1-OF3	Multiobjective asymmetric pick-up route-based fuel consumption vehicle routing problem											
A-B-CD	1.00	2.00	2.00	3.00	4.00	4.00	2.00	1.00	3.00	4.00	3.00	1.00
A-C-BD	2.50	2.00	4.00	4.00	4.00	3.00	2.50	1.00	2.00	1.00	3.00	1.00
A-D-BE	2.00	2.00	3.00	4.00	4.00	4.00	3.00	3.00	2.00	1.00	1.00	1.00
A-E-BD	1.00	3.00	1.00	3.00	2.00	3.00	4.00	1.00	2.00	2.00	4.00	4.00
B-C-AD	3.00	2.00	3.00	1.00	4.00	2.00	2.00	1.00	1.00	4.00	3.00	4.00
B-D-AC	2.00	4.00	2.00	4.00	1.00	3.00	3.00	3.00	1.00	1.00	2.00	4.00
B-E-AD	4.00	4.00	1.00	3.00	2.00	2.00	1.50	1.00	4.00	1.50	3.00	3.00
C-D-AE	1.00	1.00	2.00	3.00	4.00	4.00	4.00	2.00	3.00	2.00	3.00	1.00
C-E-AB	2.50	1.00	3.00	1.00	2.00	4.00	4.00	3.00	1.00	2.50	4.00	2.00
D-E-BC	3.00	2.00	4.00	4.00	1.00	3.00	2.00	3.00	1.00	1.00	4.00	2.00
Total rank	108.0000	101.0000	112.0000	116.5000	112.5000	114.0000	115.0000	74.5000	96.5000	60.5000	112.0000	77.5000
Average rank	2.70	2.53	2.80	2.91	2.81	2.85	2.88	1.86	2.41	1.51	2.80	1.94

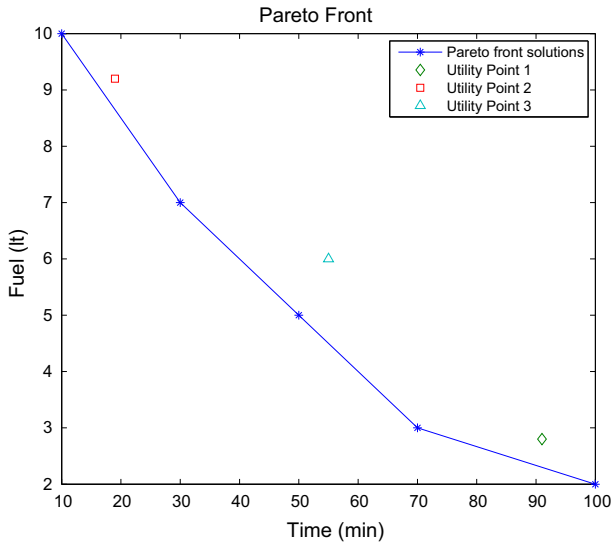


Fig. 4 Utility points on a Pareto front

$UP(Fuel) = (\text{Max Fuel value} - \text{Min Fuel value}) * (1 - UV(Fuel)) + \text{Min Fuel value} = (10 - 2) * 0.1 + 2 = 2.8$

The *Utility Point* in that case is the point (91,2.8) (*Utility Point 1*). The last step is to find based on the Euclidean distances from the real points of the Pareto Front which solution is closer to the *UP*. This solution [in our example the solution (100,2)] is the solution that the decision maker must choose. If the *Utility Variables* were $UV(Fuel) = 0.1$ and $UV(Time) = 0.9$ (*Utility Point 2*) the solution that the decision maker must choose is the solution (10,10) while if the *Utility Variables* were $UV(Fuel) = 0.5$ and $UV(Time) = 0.5$ (*Utility Point 3*) the best solution is the solution (50,5).

7 Conclusions and future research

In this paper, as we could not find DE implementations for the solution of multiobjective energy vehicle routing problems in the literature (at least to our knowledge) and we would like to see its performance in a difficult multiobjective fuel consumption vehicle routing problem we have proposed a hybridized version of a Multiobjective DE for the solution of this problem. The main contribution of the paper, in addition to the proposed algorithm, is the four new multiobjective formulations of the problem using combinations of symmetric and asymmetric formulations in pick-up and delivery problems. The formulation includes real route parameters, as the direction of the wind and the behavior of the drivers, and, thus, very realistic formulations were produced. In general, in the four different problems that are presented in this paper the three versions of DE and the hybridized version of NSGA II gave equally good results. However, the behavior of the algorithms was slightly different when a symmetric and

an asymmetric case was solved, as it was presented and analyzed in the computational results section. Our future research will be, mainly, focused on the application of those algorithms in other multiobjective combinatorial optimization problems.

References

1. Abbass, H.A.: A memetic pareto evolutionary approach to artificial neural networks. The Australian joint conference on artificial intelligence **2256**, 1–12 (2001)
2. Abbass, H.A.: The self-adaptive pareto differential evolution algorithm. In: IEEE Congress on evolutionary computation (CEC2002), vol. 1, pp. 831–836 (2002)
3. Abbass, H.A., Sarker, R.: The pareto differential evolution algorithm. *Int. J. Artif. Intell. Tools* **11**(4), 531–552 (2002)
4. Abbass, H. A., Sarker, R., Newton, C.: PDE: a Pareto-frontier differential evolution approach for multi-objective optimization problems. In: IEEE proceedings of the congress on evolutionary computation 2001 (CEC2001), vol. 2, pp. 971–978 (2001)
5. Bandeira, J.M., Fontes, T., Pereira, S.R., Fernandes, P., Khattak, A., Coelho, M.C.: Assessing the importance of vehicle type for the implementation of eco-routing systems. *Transp. Res. Proc.* **3**, 800–809 (2014)
6. Bektas, T., Laporte, G.: The pollution-routing problem. *Transp. Res. Part B* **45**, 1232–1250 (2011)
7. Chand, P., Mohanty, J.R.: Multi objective genetic approach for solving vehicle routing problem. *Int. J. Comput. Theory Eng.* **5**(6), 846–849 (2013)
8. Chakraborty, U.K.: Advances in differential evolution, *Studies in computational intelligence*, vol. 143. Springer-Verlag, Berlin, Heidelberg (2008)
9. Chang, C.S., Xu, D.Y., Quek, H.B.: Pareto-optimal set based multiobjective tuning of fuzzy automatic train operation for mass transit system. *IEE Proc. Electr. Power Appl.* **146**(5), 577–583 (1999)
10. Charoenroop, N., Satayopas, B., Eungwanichayapant, A.: City bus routing model for minimal energy consumption. *Asian J. Energy Environ.* **11**(01), 19–31 (2010)
11. Christofides, N., Mingozzi, A., Toth, P.: The vehicle routing problem. In: Christofides, N., Mingozzi, A., Toth, P., Sandi, C. (eds.) *Combinatorial Optimization*. Wiley, Chichester (1979)
12. Cicero-Fernandez, P., Long, J.R., Winer, A.M.: Effects of grades and other loads on on-road emissions of hydrocarbons and carbon monoxide. *Air and Waste Management Association* **47**, 898–904 (1997)
13. Coello Coello, C.A., Van Veldhuizen, D.A., Lamont, G.B. (2007) *Evolutionary Algorithms for Solving Multi-Objective Problems*, Springer
14. Cullen, S.: Trees and wind: Wind scales and speeds. *Journal of Arboriculture* **28**(5), 237–242 (2002)
15. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: Schoenauer, M., Deb, K., Rudolph, G., Yao, X., Lutton, E., Merelo, J.J., Schwefel, H.-P. (eds.) *Proceedings of the parallel problem solving from nature VI conference, LNCS 1917*, pp. 849–858 (2000)
16. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **6**(2), 182–197 (2002)
17. Demir, E., Bektas, T., Laport, G.: The bi-objective pollution-routing problem. *Eur. J. Oper. Res.* **232**, 464–478 (2014)
18. Demsar, J.: Statistical comparisons of classifiers over multiple data sets. *Mach. Learn. Res.* **7**, 1–30 (2006)
19. Engelbrecht, A.P.: *Computational intelligence: an introduction*. Wiley, England (2007)
20. Erbao, C., Mingyong, L.: A hybrid differential evolution algorithm to vehicle routing problem with fuzzy demands. *J. Comput. Appl. Math.* **231**(1), 302–310 (2009)
21. Erbao, C., Mingyong, L.: The open vehicle routing problem with fuzzy demands. *Expert Syst. Appl.* **37**(3), 2405–2411 (2010)
22. Erbao, C., Mingyong, L., Yang, H.: Open vehicle routing problem with demand uncertainty and its robust strategies. *Expert Syst. Appl.* **41**(7), 3569–3575 (2014)
23. Erdogan, S., Miller-Hooks, E.: A green vehicle routing problem. *Transp. Res. Part E* **48**, 100–114 (2012)
24. Feo, T.A., Resende, M.G.C.: Greedy randomized adaptive search procedure. *J. Glob. Optim.* **6**, 109–133 (1995)

25. Figliozzi, M.: Vehicle routing problem for emissions minimization. *Transp. Res. Rec. J Transp. Res. Board* **2**, 1–7 (2011)
26. Friedman, M.: The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *J. Am. Stat. Assoc.* **32**, 675–701 (1937)
27. Friedman, M.: A comparison of alternative tests of significance for the problem of m rankings. *Ann. Math. Stat.* **11**, 86–92 (1940)
28. Garcia, S., Herrera, F.: An extension on 'Statistical comparisons of classifiers over multiple data sets' for all pairwise comparisons. *Mach. Learn. Res.* **9**, 2677–2694 (2008)
29. Hansen, P., Mladenovic, N.: Variable neighborhood search: Principles and applications. *Eur. J. Oper. Res.* **130**, 449–467 (2001)
30. Huayu, X., Wenhui, F., Tian, W., Lijun, Y.: An or-opt NSGA-II algorithm for multi-objective vehicle routing problem with time windows. In: 4th IEEE conference on automation science and engineering, pp. 309–314 (2008)
31. Iman, R.L., Davenport, J.M.: Approximations of the critical region of the Friedman statistic. In: *Communications in statistics*, pp. 571–595 (1980)
32. Iorio, A.W., Li, X.: Solving rotated multi-objective optimization problems using differential evolution. In: *AI 2004: advances in artificial intelligence*, vol. 3339, pp. 861–872 (2004)
33. Jemai, J., Zekri, M., Mellouli, K.: An NSGA-II algorithm for the green vehicle routing problem. *Evol. Comput. Comb. Optim. Lect. Notes Comput. Sci.* **7245**, 37–48 (2012)
34. Johnson, D.S., Papadimitriou, C.H.: Computational complexity. In: Lawer, E.L., Lenstra, J.K., Rinnoy Kan, A.H.D., Shmoys, D.B., (eds.) *The traveling salesman problem: a guided tour of combinatorial optimization*, pp. 37–85. New York, Wiley (1985)
35. Jozefowicz, N., Semet, F., Talbi, E.G.: Enhancements of NSGA II and its application to the vehicle routing problem with route balancing. *Artif. Evol. Lect. Notes Comput. Sci.* **3871**, 131–142 (2006)
36. Jozefowicz, N., Semet, F., Talbi, E.G.: Multi-objective vehicle routing problems. *Eur. J. Oper. Res.* **189**, 293–309 (2008)
37. Kara, I., Kara, B.Y., Yetis, M.K.: Energy minimizing vehicle routing problem. *COCOA* **2007**, 62–71 (2007)
38. Koc, C., Bektas, T., Jabali, O., Laporte, G.: The fleet size and mix pollution-routing problem. *Transp. Res. Part B* **70**, 239–254 (2014)
39. Kontovas, C.A.: The green ship routing and scheduling problem (GSRSP): A conceptual approach. *Transp. Res. Part D* **31**, 61–69 (2014)
40. Kukkonen, S., Lampinen, J.: An extension of generalized differential evolution for multi-objective optimization with constraints. In: *Parallel Problem Solving from Nature—PPSN VIII, LNCS*, vol. 3242, pp. 752–761 (2004)
41. Kumar, R.S., Kondapaneni, K., Dixit, V., Goswami, A., Thakur, L.S., Tiwari, M.K.: Multi-objective modeling of production and pollution routing problem with time window: a self-learning particle swarm optimization approach. *Comput. Ind. Eng.* (pii:S0360–8352(15)00287–9) (2015)
42. Kuo, Y.: Using simulated annealing to minimize fuel consumption for the time-dependent vehicle routing problem. *Comput. Ind. Eng.* **59**(1), 157–165 (2010)
43. Labadie, N., Prodhon, C.: A survey on multi-criteria analysis in logistics: Focus on vehicle routing problems. In: *Applications of multi-criteria and game theory approaches springer series in advanced manufacturing*, pp. 3–29 (2014)
44. Lahyani, R., Khemakhem, M., Semet, F.: Rich vehicle routing problems: from a taxonomy to a definition. *Eur. J. Oper. Res.* **241**, 1–14 (2015)
45. Laporte, G.: The vehicle routing problem: An overview of exact and approximate algorithms. *Eur. J. Oper. Res.* **59**, 345–358 (1992)
46. Lawer, E.L., Lenstra, J.K., Rinnoy Kan, A.H.G.R., Shmoys, D.B.: *The traveling salesman problem: a guided tour of combinatorial optimization*. New York, Wiley (1985)
47. Leonardi, J., Baumgartner, M.: *C O₂ efficiency in road freight transportation: Status quo, measures and potential*. *Transp. Res. Part D* **9**, 451–464 (2004)
48. Li, H., Lv, T., Li, Y.: The tractor and semitrailer routing problem with many-to-many demand considering carbon dioxide emissions. *Transp. Res. Part D* **34**, 68–82 (2015)
49. Li, J.: Vehicle routing problem with time windows for reducing fuel consumption. *J. Comput.* **7**(12), 3020–3027 (2012)
50. Lin, C., Choy, K.L., Ho, G.T.S., Ng, T.W.: A genetic algorithm-based optimization model for supporting green transportation operations. *Expert Syst. Appl.* **41**, 3284–3296 (2014)

51. Lin, C., Choy, K.L., Ho, G.T.S., Chung, S.H., Lam, H.Y.: Survey of green vehicle routing problem: Past and future trends. *Expert Syst. Appl.* **41**(4), 1118–1138 (2014)
52. Lin, S.: Computer solutions of the traveling salesman problem. *Bell Syst. Tech. J.* **44**, 2245–2269 (1965)
53. Marinakis, Y., Iordanidou, G.R., Marinaki, M.: Particle swarm optimization for the vehicle routing problem with stochastic demands. *Appl. Soft Comput.* **13**, 1693–1704 (2013)
54. Marinakis, Y., Marinaki, M., Spanou P.: A memetic differential evolution algorithm for the vehicle routing problem with stochastic demands. In: Fister, I., Fister, I., Jr. (eds.) *Adaptation and hybridization in computational intelligence, adaptation, learning, and optimization*, vol. 18, pp. 185–203 (2015)
55. McKinnon, A.: A logistical perspective on the fuel efficiency of road freight transport. In: *Workshop Proceedings, Paris, OECD, ECMT and IEA* (1999)
56. McKinnon, A.: Green logistics: the carbon agenda. *Electron. Sci. J. Logist.* **6**(3), 1–9 (2010)
57. Mezura-Montes, E., Reyes-Sierra, M., Coello Coello, C.: Multi-objective optimization using differential evolution: a survey of the state-of-the-art. In: *Advances in differential evolution, studies in computational intelligence*, vol. 143. Springer-Verlag, Berlin, Heidelberg (2008)
58. Mingyong, L., Erbao, C.: An improved differential evolution algorithm for vehicle routing problem with simultaneous pickups and deliveries and time windows. *Eng. Appl. Artif. Intell.* **23**(2), 188–195 (2010)
59. Molina, J.C., Eguia, I., Racero, J., Guerrero, F.: Multi-objective vehicle routing problem with cost and emission functions. *Proc. Soc. Behav. Sci.* **160**, 254–263 (2014)
60. Okabe, T., Jin, Y., Sendhoff, B.: A critical survey of performance indices for multi-objective optimization. *Evol. Comput.* **2**, 878–885 (2003)
61. Ombuki, B., Ross, B.J., Hanshar, F.: Multi-objective genetic algorithms for vehicle routing problem with time windows. *Appl. Intell.* **24**, 17–30 (2006)
62. Parsopoulos, K.E., Tasoulis, D.K., Pavlidis, N.G., Plagianakos, V.P., Vrahatis, M.N.: Vector evaluated differential evolution for multiobjective optimization. In: *2004 Congress on evolutionary computation (CEC 2004)*, Portland, June 2004, vol. 1, pp. 204–211, IEEE Service Center (2004)
63. Price, K.V., Storn, R.M., Lampinen, J.A.: *Differential evolution: a practical approach to global optimization*. Springer, Berlin (2005)
64. Psychas, I.D., Delimpasi, E., Marinakis, Y.: Hybrid evolutionary algorithms for the multiobjective traveling salesman problem. *Expert Syst. Appl.* doi:[10.1016/j.eswa.2015.07.051](https://doi.org/10.1016/j.eswa.2015.07.051) (2015, in press)
65. Psychas, I.D., Marinaki, M., Marinakis, Y.: A parallel multi-start NSGA II algorithm for multiobjective energy reduction vehicle routing problem. In: Gaspar-Cunha, A. et al. (eds.) *8th International conference on evolutionary multicriterion optimization, EMO 2015, Part I, LNCS 9018*, Springer International Publishing Switzerland 2015, pp. 336–350 (2015)
66. Robic, T., Filipic, B.: DEMO: differential evolution for multiobjective optimization, evolutionary multi-criterion optimization. In: *Third International Conference, EMO 2005*, vol. 3410, pp. 520–533 (2005)
67. Santana-Quintero, L.V.: Coello Coello, C.A.: An algorithm based on differential evolution for multi-objective problems. *Int. J. Comput. Intell. Res.* **1**(2), 151–169 (2005)
68. Sarker, R., Coello Coello, C.A.: Assessment methodologies for multiobjective evolutionary algorithms. In: *Evolutionary optimization, international series in operations research and management science*, vol. 48, pp. 177–195 (2002)
69. Sauer, J. G.: Discrete differential evolution with local search to solve the traveling salesman problem: fundamentals and case studies. In: *Cybernetic intelligent systems*, pp. 1–6 (2008)
70. Sbihi, A., Eglese, R.W.: Combinatorial optimization and green logistics. *4OR* **5**(2), 99–116 (2007)
71. Schaffer, J.: Some experiments in machine learning using vector evaluated genetic algorithms, PhD thesis, Vanderbilt University (1984)
72. Silva, A. L., Ramirez, J. A., Campelo, F.: A statistical study of discrete differential evolution approaches for the capacitated vehicle routing problem. In: *Proceedings of the 15th annual conference companion on genetic and evolutionary computation, (GECCO '13 Companion)*, pp. 77–78 (2013)
73. Storn, R., Price, K.: Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **11**(4), 341–359 (1997)
74. Suzuki, Y.: A new truck-routing approach for reducing fuel consumption and pollutants emission. *Transp. Res. Part D* **16**, 73–77 (2011)

75. Tajik, N., Tavakkoli-Moghaddam, R., Vahdani, B.: Meysam Mousavi, S.: A robust optimization approach for pollution routing problem with pickup and delivery under uncertainty. *J. Manuf. Syst.* **33**, 277–286 (2014)
76. Tiwari, A., Chang, P.C.: A block recombination approach to solve green vehicle routing problem. *Int. J. Prod. Econ.*, pp. 1–9 (2002)
77. Toth, P., Vigo, D.: The vehicle routing problem. *Monographs on Discrete Mathematics and Applications*, Siam (2002)
78. Toth, P., Vigo, D.: Vehicle routing: Problems, methods and applications, MOS-Siam Series on Optimization, 2nd edn. Siam, Philadelphia (2014)
79. Weizhen, R., Chun, J.: A model of vehicle routing problem minimizing energy consumption in urban environment. In: Asian conference of management science & applications, September 2012, Chengdu-Jiuzhaigou, pp. 21–29 (2012)
80. Xiao, Y., Zhao, Q., Kaku, I., Xu, Y.: Development of a fuel consumption optimization model for the capacitated vehicle routing problem. *Comput. Oper. Res.* **39**(7), 1419–1431 (2012)
81. Xue, F.: Multi-objective differential evolution: theory and applications, PhD thesis, Rensselaer Polytechnic Institute, New York (2004)
82. Xue, F., Sanderson, A.C., Graves, R.J.: Pareto-based multiobjective differential evolution. In: IEEE Proceedings of the 2003 congress on evolutionary computation (CEC2003), vol. 2, pp. 862–869 (2003)
83. Zhang, S., Lee, C.K.M., Choy, K.L., Ho, W., Ip, W.H.: Design and development of a hybrid artificial bee colony algorithm for the environmental vehicle routing problem. *Transp. Res. Part D* **31**, 85–99 (2014)
84. Zitzler, E., Deb, K., Thiele, L.: Comparison of multiobjective evolutionary algorithms: Empirical results. *Evol. Comput.* **8**(2), 173–195 (2000)