

# Orbit Transfer Manoeuvres as a Test Benchmark for Comparison Metrics of Evolutionary Algorithms

E.A. Minisci and G. Avanzini

**Abstract**—In the present paper some metrics for evaluating the performance of evolutionary algorithms are considered. The capabilities of two different optimisation approaches are compared on three test cases, represented by the optimisation of orbital transfer trajectories. The complexity of the problem of ranking stochastic algorithms by means of quantitative indices is analyzed by means of a large sample of runs, so as to derive statistical properties of the indices in order to evaluate their usefulness in understanding the actual algorithm capabilities and their possible intrinsic limitations in providing reliable information.

## I. INTRODUCTION

Orbit transfer manoeuvres provide a demanding test benchmark for evolutionary optimisation algorithms. Even in the simplest scenarios, the resulting optimisation problem can be multi-modal, with a Pareto front in the fuel-consumption vs transfer time plane which may prove difficult to explore. At the same time, as far as the application of evolutionary techniques in the framework of space mission analysis and design is concerned, it is of paramount importance to understand the capabilities of a given (class of) optimization algorithm(s), as compared to others, for a certain (class of) orbit transfer manoeuvre(s) [1].

The importance of achieving a prescribed orbit within a given time while minimizing the amount of fuel used is somehow self evident. Any saving on fuel required for a given mission phase results into either a longer operational life, a higher available payload for the same launch weight, a smaller total launch weight, or any combination of the above benefits. The capabilities of the computational tools adopted for mission design have thus a direct impact on shaping the mission, and choosing the best suited ones is an issue to be appropriately addressed in the very initial phases of the mission design process. The present paper is aimed at discussing some performance metrics in order to evaluate and compare the capabilities of different evolutionary optimisation approaches to the same problem.

Evolutionary algorithms include a very wide range of numerical procedures, that range from genetic algorithms (GAs) [2] to evolutionary strategies (ESs) [3] and differential evolution (DE) [4]. They represent a powerful global optimization method, and, although convergence to a global optimum is only guaranteed in a weak probabilistic sense,

they are well suited for a wide range of both combinatorial and continuous problems. Different approaches may achieve different performance depending on the considered problem, so that a systematic comparison of algorithm performance is extremely interesting. Unfortunately, because of the stochastic nature of the algorithms, performance indices must be considered as aleatory variables. A comparison becomes thus computationally demanding, requiring a very large number of runs for each one of the approaches in order to characterize the statistical properties of the considered metrics.

Even though several papers compared different EAs [5], [6], only few of them took into consideration the statistical nature of performance indices. A first attempt to identify those global optimization algorithms that on average outperform all others over different classes of orbit transfer problems (namely two-impulse transfers, low-thrust transfer and low-energy transfers in the framework of the restricted 3-body problem) was developed by Di Lizia *et al.* [1], demonstrating that DE methods perform well on most of the problems, compared to other algorithms. More recently, Vasile *et al.* [7] proposed some performance metrics for comparing global optimization methods, discussing the actual significance of the considered indices and proposing some criteria for evaluating the usefulness of each algorithm. They addressed specifically black-box problems in space trajectory design, focusing their attention on stochastic approaches. The metrics used for evaluating algorithm performance were based on the capabilities demonstrated by different approaches over several runs, repeatability of the results being one of the most relevant aspects. They considered single-objective optimizations, but the extension to multi-objective (MO) problems is far from trivial. In this case the computation effort for a reliable evaluation of performance metrics can become overwhelming, as the solution is not a single optimal solution but a whole population spread along the Pareto front [8]. Since it is not possible to perform an infinite number of runs for each case, it is important to estimate the error on the indices computed for a limited number of tests, which can be very low, depending on the available computational resources.

In their preliminary study [8], Minisci and Avanzini used only 20 runs for each case for comparing the performance of two evolutionary algorithms, the NSGA-II [9] and the MOPED [10] algorithms. The results were analyzed in terms of confidence level, where the definition of successful runs as those runs where the performance index is better than a predetermined threshold leads to a binomial probability density function, independently of the number of function

E.A. Minisci is with the Department of Aerospace Engineering, University of Glasgow, J. Watt (South) Building, G12 8QQ Glasgow, Scotland (phone: +44 (0)141 3304470; fax: +44 (0)141 3305560; email: eminisci@eng.gla.ac.uk). G. Avanzini is with the Department of Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy.

evaluations, problem formulation and optimization approach. As a major advantage, Vasile and his co-workers cleverly pointed out that this method allows for designing the test knowing *a priori* the relation between the number of runs and the error on the estimation of the success index [7]. This work is aimed at extending the previous study by the same authors, by means of a more detailed analysis of the statistical properties of the performance indices, performed on a set of 200 runs over 3 different cases. Nonetheless, the identification of the “best” algorithm remains questionable, in the presence of different (and often competing) metrics (accuracy of the solution, capability of converging onto the global optimum, repeatability of the solution, and so on), no matter how accurate and extensive the statistical analysis [8], especially when MO problems are dealt with. This means that, in spite of the computation effort at the basis of the reported results, it is still hard to draw final conclusions, although a more in-depth insight on algorithm performance is now available.

In the framework of space mission analysis applications, Minisci and Avanzini showed how algorithm performance are affected by the representation of the orbital problem, because of the characteristics of the resulting search space and its functional relation with the objective function(s) [8]. In this latter respect, the use of inverse method, where the transfer orbit is identified by means of the solution of the Lambert problem [11], usually provides better results than those obtained by means of direct orbit propagation, where constraints must be enforced on the acquisition of the final target orbit [8]. At the same time, this latter technique challenges more seriously the optimisation algorithm, especially as far as the search of feasible solution is concerned, so that this approach will be considered as one of the benchmark cases for comparing the NSGA-II and the MOPED algorithms.

The two optimization algorithms are briefly presented in the next Section (more details can be found in [9], [10] and [12]). In Section III, after presenting problem geometry, objectives, constraints and benchmark manoeuvres (a 2-impulse transfer from LEO to a Molniya-like orbit and a 3-impulse transfer from LEO to GEO), two different orbit transfer parameterizations are discussed: the first one is based on the use of velocity increment components as optimization variables; in the second case the transfer orbit is determined by means of the solution of the Lambert problem [11], [13], where the considered initial and final positions represent the optimization variables. The metrics used for comparing the optimisation algorithm performance are then presented in Section IV, while Section V reports the set of results obtained by the two optimisation methods on the benchmark manoeuvres. A Section of Conclusions ends the paper.

## II. EVOLUTIONARY OPTIMIZATION METHODS

Two multi-objective optimization algorithm are compared in this paper. The first one is a standard Genetic Algorithm (GA), the NonDominated Sorting Genetic Algorithm-II (NSGA-II), which still represents one of the best multi-objective GA (MOGA) available [9]. The second one is an

Evolutionary Algorithm (EA) which belongs to the subclass of Estimation of Distribution Algorithms (EDAs) [10]. A brief description of both algorithms is here presented, pointing out similarities and main differences, in order to allow the reader to better understand the analysis of the reported results.

### A. Non-Dominated Sorting Genetic Algorithm-II (NSGA-II)

In the NSGA-II the dominance depth is used to classify the population. A crowding parameter is also determined in order to rank the individuals inside each class of dominance. For each element of a class, the crowding parameter is obtained as the sum of the difference of the cost functions of the nearest elements in the cost function space, divided by the range spanned by the population with respect to each objective function. Inside each class, the individuals with the higher value of the crowding parameter obtain a better rank than those with a lower one, forcing to explore the Pareto front.

For unconstrained optimization problems, the evolution is started from a random parent population,  $\mathcal{P}_0$ . The population is sorted by means of the non-domination criterion, whereas each individual is assigned a fitness parameter equal to its non-domination level. A fitness value equal to 1 (best cases) is assigned to the non-dominated individuals, that form the first layer. Those individuals dominated only by members of the first layer form the second one and are assigned a fitness value of 2, and so on. In general, for dominated individuals, the fitness is given by the number of dominating layers plus 1 and minimization of fitness index is pursued by the algorithm.

Binary tournament selection, recombination, and mutation operators [9] are used to create a child population  $\mathcal{Q}_0$  of size  $N_{ind}$  from the initial generation. Given a parent and a child generation, indicated by the subscript  $t \geq 0$ , the procedure generates a combined population  $\mathcal{R}_t = \mathcal{P}_t \oplus \mathcal{Q}_t$  of size  $2N_{ind}$ . Then,  $\mathcal{R}_t$  is sorted according to the non-domination criterion. The new parent population  $\mathcal{P}_{t+1}$  of size  $N_{ind}$  is formed by first sorting the members of the last front according to the crowding comparison operator and then picking up the first  $N_{ind}$  individuals that form  $\mathcal{P}_{t+1}$ . The new parent population can now be used for selection, crossover and mutation to create a new child population  $\mathcal{Q}_{t+1}$  of size  $N_{ind}$  and iterate the process.

In this work a real-coded, constrained version (simulated binary crossover, SBX, and polynomial mutation) has been used. Constraints are handled according to the constraint-domination principle [9], which discriminates between unfeasible and feasible solutions during the non-dominated sorting procedure. The definition of constrained-domination is: a solution  $i$  is said to constrained-dominate a solution  $j$ , if (a) solution  $i$  is feasible and solution  $j$  is unfeasible; or (b) solution  $i$  and  $j$  are both unfeasible, but solution  $i$  is closer to the constraint boundary; or (c) if both solutions are feasible but  $i$  dominates  $j$  with respect to the objectives functions.

The parameters to be set are the size of the population,  $N_{ind}$ , the crossover and mutation probability,  $p_c$  and  $p_m$ , and distribution indices for crossover and mutation,  $\eta_c$  and  $\eta_m$ , respectively.

## B. Multi-Objective Parzen-Based Estimation of Distribution Algorithm (MOPED)

The MOPED algorithm is a multi-objective optimization algorithm for continuous problems that uses the Parzen method to build a probabilistic representation of Pareto solutions, with multivariate dependencies among variables. The Parzen method [10] pursues a non-parametric approach to kernel density estimation and it gives rise to an estimator that converges everywhere to the true Probability Density Function (PDF) in the mean square sense. Should the true PDF be uniformly continuous, the Parzen estimator can also be made uniformly consistent. In short, the method allocates exactly  $N_{ind}$  identical kernels, each one centered on a different element of the sample.

Similarly to what was done in [14] for multi-objective Bayesian Optimization Algorithm (moBOA), some techniques of NSGA-II are used to classify promising solutions in the objective space, while new individuals are obtained by sampling from the Parzen model. NSGA-II was identified as a promising base for the algorithm mainly because of its intuitive simplicity coupled with brilliant results on many problems. The major differences between MOPED and NSGA-II, due to the classification and search techniques, are here summarized.

1) *Classification and Fitness evaluation:* The individuals of the population are classified in a way that favors the most isolated individuals in the objective function space, in the first sub-class (highest dominance) of the first class (best suited with respect to problem constraints).

If the problem is characterized by  $m$  constraints  $c_i(\mathbf{x})$ ,  $i = 1, 2, \dots, m$ , such that  $c_j(\mathbf{x}) = 0$  indicates that the  $j$ -th constraint is satisfied, the first step in the evaluation of the fitness parameter is the determination of the degree of compatibility of each individual with the constraints. The compatibility, indicated by the symbol  $cp$ , is measured as the weighted sum of unsatisfied constraint. Once the value of  $cp$  is evaluated for all the individuals, the population is distributed over a predetermined number of classes,  $1 + N_{cl}$ . The  $N_{best}$  individuals that satisfy all the constraints, such that  $cp = 0$ , are in the first class. The remainder of the population is divided in the remaining  $N_{cl}$  groups, each one containing an approximately equal number of individuals, given by  $\text{round}((N_{ind} - N_{best})/N_{cl})$ .

The second class is formed by those individuals with the lower values of the constraint parameter and the last one by those with the highest values. For each class, individuals are ranked in terms of dominance criterion and crowding distance in the objective function space, using the NSGA-II techniques. After ranking all the individuals of the population, from the best to the worst one, depending on their belonging to a given class and dominance level and the value of their crowding parameter, a fitness value  $f$  linearly varying from  $2 - \alpha$  (best individual of the entire population) to  $\alpha$  (worst individual), with  $\alpha \in [0; 1)$ , is assigned to each individual. This fitness value determines the weighting of the kernel for sampling the individuals of the next generation. As an

example, for  $\alpha = 0$ , the best solution ( $f = 2$ ) provides a kernel with twice as much possibilities of generating new individuals for the next generation than the central one, placed at half of the classification (for a corresponding value of  $f = 1$ ), while the kernel for the worst one ( $f = 0$ ) is prevented from generating new individuals. Higher values of  $\alpha$  are usually employed for allowing sampling of regions of the search space far from the current best solutions.

2) *Building the model and sampling:* As briefly outlined in the previous section, a probabilistic model of the promising search space portion is built on the basis of the information given by  $N_{ind}$  individuals of the current population, by means of the Parzen method. On the basis of this model,  $\tau N_{ind}$  new individuals, with  $\tau \geq 1$  are sampled. The variance associated to each kernel depends on (i) the distribution of the individuals in the search space and (ii) on the fitness value associated to the pertinent individual, so as to favor sampling in the neighborhood of the most promising solutions. For generic processes it can be useful to adopt different kernels alternatively, going from one generation to the following one, in order to improve the exploration of the search space.

The parameters to be set for the MOPED algorithm are: size of the population,  $N_{ind}$ , number of constraint classes,  $N_{cl}$ , the fitness coefficient,  $\alpha$ , the sampling proportion,  $\tau$ .

## III. PROBLEM STATEMENT

### A. Geometry, Objectives and Constraints

A generic  $N$ -impulse orbit transfer requires the definition of  $N$  velocity increments,  $\Delta\vec{v}_i$ ,  $i = 1, 2, \dots, N$ , that take a spacecraft from an initial orbit to a given target one along a trajectory formed by  $N - 1$  coast arcs. Each velocity increment  $\Delta\vec{v}_i$  is identified by means of its magnitude  $\Delta v_i = \|\Delta\vec{v}_i\|$  and angle with respect to the tangential direction  $\phi_i$  along the current coast arc. The first impulse at time  $t_1 = t_W$  (where  $t_W$  indicates the waiting time on the initial orbit) injects the spacecraft on the first coast arc. The last coast arc must intersect the target orbit and the last impulse is required to inject the spacecraft on the target orbit, thus completing the prescribed maneuver. The parameters of each coast arc will be identified by the subscript  $T_i$ ,  $i = 1, 2, \dots, N - 1$ .

The initial and target orbits are defined by means of their orbital elements, indicated by the subscript  $I$  and  $F$ , respectively. In the present work coplanar orbit transfer starting from a circular LEO will be dealt with, in which case problem geometry is defined by 5 quantities only, namely semi-major axis,  $a_I$  and  $a_F$ , eccentricity,  $e_I$  and  $e_F$ , and initial spacecraft position on the initial orbit,  $\theta_{\sigma_0}$ . The periapsis of the target orbit is assumed as reference for the anomalies. If both orbits are circular, spacecraft position at injection time has no influence on the transfer, so that  $t_1 = t_W = 0$  and  $\theta_{\sigma_0} = 0$  are assumed without loss of generality.

The search for minimum-fuel/minimum-time transfer results in a multi-objective optimization problem, where the magnitude of the velocity increment, related to the fuel necessary to perform the transfer, and the total time required

to complete the transfer, are given by

$$\Delta V_{tot} = \sum_{i=1}^N \|\Delta \vec{v}_i\|, \quad T_{tot} = t_W + \sum_{i=1}^{N-1} \Delta t_i \quad (1)$$

respectively,  $\Delta t_i$  representing the transfer time along the  $i$ -th coast arc.

If a rendezvous problem is dealt with, a further problem parameter is the target position at the initial time,  $\theta_{\tau_0}$ . In such a case a constraint on the final spacecraft position is represented by the condition  $\theta_{\sigma_F} = \theta_{\tau_F}$ , where the subscripts  $\sigma$  and  $\tau$  indicate spacecraft and target, respectively.

Inequality constraints are also present. In particular, only arcs of elliptic orbits will be considered as admissible transfer trajectories and each arc must remain higher than a prescribed minimum altitude over the surface of the planet, assumed equal to the altitude of the initial LEO of radius  $r_I$ . Two sets of inequality constraints result, in the form

$$e_{T,i} < 1, \quad r_{T,i,min} \geq r_I, \quad i = 1, 2, \dots, N-1. \quad (2)$$

where  $r_{T,i,min}$  is the minimum radius along the  $i$ -th transfer orbit arc.

### B. Representation of Orbit Transfers

As highlighted in [8], the representation of the same orbit transfer by means of different parametrizations affects the convergence capabilities of the optimisation algorithm, because of the different features of the search space and functional dependencies of the merit functions from the optimisation variables. The two representations considered for the orbit transfer problem for Keplerian motion are (i) a standard orbit propagation algorithm ( $f$ - $g$  method based on Lagrangian coefficients [11]) and (ii) the solution of the Lambert problem.

1) *Orbit propagation*: When orbit propagation is used for determining the optimal  $N$ -impulse transfer, the optimization variables are magnitude and direction of the velocity increments, namely  $\Delta v_i$  and  $\phi_i$ ,  $i = 1, 2, \dots, N$ , together with the anomalies  $\theta_i$  of the corresponding positions, for a total of  $3N$  design variables; 4 equality constraints on semi-major axis,  $a_F$ , eccentricity,  $e_F$ , argument of the periapsis,  $\omega_F$ , and position along the orbit,  $\theta_\sigma = \theta_\tau$  at  $t = t_F$ , for acquisition of the target on the final orbit need to be enforced together with two inequality constraints for each arc of the transfer trajectory, Eq. (2).

Each equality constraint on orbit parameters and spacecraft final anomaly (rendezvous condition) is enforced by means of two inequality constraints in the form

$$\begin{aligned} a_{F,des} - \varepsilon_a &\leq a_F \leq a_{F,des} + \varepsilon_a \\ e_{F,des} - \varepsilon_e &\leq e_F \leq e_{F,des} + \varepsilon_e \\ \omega_{F,des} - \varepsilon_\omega &\leq \omega_F \leq \omega_{F,des} + \varepsilon_\omega \\ \theta_{F,des} - \varepsilon_\theta &\leq \theta_F \leq \theta_{F,des} + \varepsilon_\theta \end{aligned} \quad (3)$$

where  $\varepsilon_a$ ,  $\varepsilon_e$ ,  $\varepsilon_\omega$ , and  $\varepsilon_\theta$  are the assumed tolerances.

TABLE I  
ORBITS FOR BENCHMARK CASES ( $\mu_\oplus = 3.9860 \cdot 10^5 \text{ km}^3\text{s}^{-2}$ )

Manoeuvre	$a_I$	$e_I$	$a_F$	$e_F$
A	6 721 km	0	26 610 km	0.667
B	7 000 km	0	42 000 km	0

2) *Lambert Problem*: The solution of the two-point boundary value problem for Keplerian motion, also known as Lambert's problem [11], is represented by the determination of the orbit parameters of an orbit having a specified transfer time  $t_{des}$  between two prescribed positions in space,  $P_1$  and  $P_2$ . In this case, each coast arc of the transfer trajectory is identified by means of the assumed initial and final positions and transfer time. The total number of design variables is thus equal to  $3(N-2) + 2$ .

The required  $\Delta V$  is evaluated from the knowledge of the orbit parameters, by means of a simple vector operation,

$$\Delta \vec{v}_i = \vec{v}_i^I - \vec{v}_{i-1}^F, \quad i = 1, 2, \dots, N$$

where  $\vec{v}_i^I$  and  $\vec{v}_i^F$  are the initial and final velocities at the edges of the  $i$ -th coast arc, for  $1 < i < N-1$ , while  $\vec{v}_0^F = \vec{v}_{LEO}$  and  $\vec{v}_N^I = \vec{v}_F$  are the velocities on the initial LEO and final target orbit, respectively.

When a Lambert algorithm is used for defining the geometrical properties of the transfer trajectory, inequality constraints apply to each one of the  $N-1$  coast arcs as in the previous case, but the constraints on the final position of the spacecraft at the end of the transfer are inherently satisfied. This fact greatly simplifies the structure of the feasible solution space, inasmuch as, together with the number of equality constraints, also the number of optimization variables is reduced. This is done at the expenses of a higher computational cost for the evaluation of a single individual of the population of candidate solutions, which requires the iterative solution of the Lambert problem for the considered set of transfer parameters (starting and arrival positions on the initial and final orbits and transfer time). In the present work, Lambert's problem is solved by means of the technique presented in [13].

### C. Benchmark manoeuvres

Two transfer manoeuvres will be considered as test-benchmarks for the considered optimization approaches, in order to highlight their performance and relevant characteristics: a two-impulse orbit transfer from LEO to a high-eccentricity Molniya-like orbit (Manoeuvre A), and a three-impulse LEO-to-GEO transfer (Manoeuvre B). The relevant data relative to these cases are summarized in Tab. 1. In all the considered cases, minimum-fuel/minimum-time solutions are sought, with rendezvous with a target spacecraft on the final orbit.

Manoeuvre A is representative of a transfer from a circular LEO to a high eccentricity semisynchronous orbit. The transfer depends on two parameters only, namely  $t_W$  and  $t_T$ . Figure 1 represents a contour plot of  $\Delta v_{tot}$  as a function of these two parameters (white  $< 3$  km/s, black  $> 15$

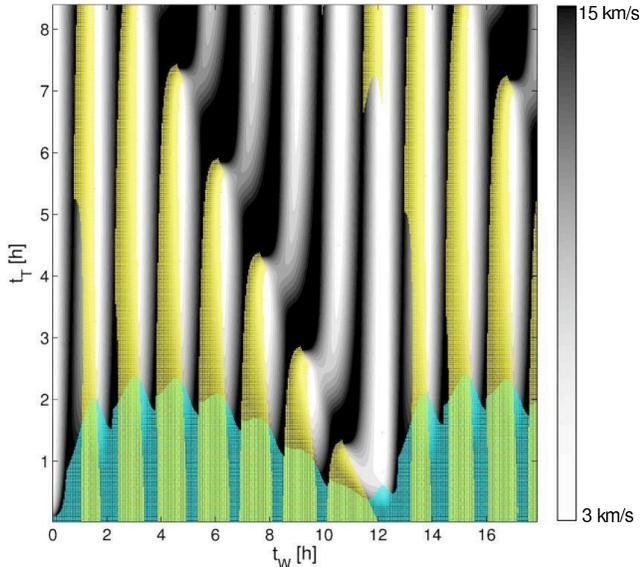


Fig. 1.  $\Delta V_{tot}$  for Manoeuvre A.

km/s). The shaded areas represent unfeasible solutions where constraints on eccentricity (cyan) and/or minimum radius (yellow) are violated. The variation of  $\Delta V_{tot}$  in the  $t_W$ - $t_T$  plane is characterized by several local minima and the shape of constraints on the transfer orbit quite complex, so that the Pareto front in the  $\Delta V_{tot}$ - $T_{tot}$  plane is expected to prove difficult to explore even in this simple scenario, where only two variables are sufficient to define the entire search space.

Manoeuvre B is a three-impulse transfer between circular orbits. Given the geometry of the initial and final orbits, a third impulse is allowed during the transfer, which is thus divided into two arcs. In such a case the solution space to be spanned becomes wider. Constraints on the eccentricity and minimum distance from the planet surface are enforced for both arcs, as for the two-impulse transfer.

Three cases will thus be considered in the present analysis, where  $\mathbf{x}$  is the vector of search variables:

- 1) Manoeuvre A parametrized by means of the solution of Lambert's problem between the initial and final positions, where only two variables are sufficient for defining the search space,  $\mathbf{x} = (t_W, t_T)^T \in \mathbb{R}^2$ , as outlined above;
- 2) Manoeuvre B parametrized by means of the solution of Lambert's problem between the initial and final position along two coast arcs,  $\mathbf{x} = (t_W, t_{T_1}, r_1, \Delta\theta_1, t_{T_2})^T \in \mathbb{R}^5$ , where  $r_1$  and  $\Delta\theta_1$  assign the position of the intermediate impulse, while  $t_{T_1}$  and  $t_{T_2}$  are the transfer times along the first and the second coast arc, respectively;
- 3) Manoeuvre A parametrized by means of an orbit propagation algorithm, where the search space is given by  $\mathbf{x} = (\theta_1, \Delta v_1, \phi_1, \Delta\theta, \Delta v_2, \phi_2)^T \in \mathbb{R}^6$ , and 4 equality constraints on the final condition, Eqs. (3), need to be enforced.

#### IV. TEST METHODOLOGY

The optimization algorithms adopted in this work belong to the wide class of stochastic algorithm. These methods are expected to converge to the global solution of the problem if the number of evaluations of the system model  $N_{eval}$  is

sufficiently high. Equivalently, letting  $P_s$  be the probability to find the global solution,  $P_s \rightarrow 1$  if  $N_{eval} \rightarrow \infty$ . It is obviously unpractical to allow an unbounded growth of  $N_{eval}$ , both for test cases and for real problems, so that the common practice is to stop the algorithm after a prescribed maximum number of evaluations of the system model,  $N_{MAX}$ . This means that, given a problem and an algorithm to solve it, it is important to evaluate algorithm effectiveness for a finite value of  $N_{MAX}$ . Because of the stochastic nature of the algorithms, their performance must be considered as an aleatory variable as well, whichever the metrics adopted to measure it.

##### A. Comparison metrics

All the benchmark problems considered in this work are constrained and multi-objective. Therefore the performance of the algorithms must be measured in terms of constraint satisfaction and approximation of the global Pareto front.

A first index gives information about the capability of the algorithm of finding at least one feasible solution. Given the total number of runs for the algorithm,  $N_{run}$ , the index  $P_{FS}$  is the ratio between the number of times the algorithm is able to find at least one feasible solution,  $N_{runF}$ , and the total number of runs:  $P_{FS} = N_{runF}/N_{run}$ .

As for the two main goals of MOEAs, that is, (i) convergence to the true Pareto optimal front, and (ii) distribution of the population over the whole front, it is necessary to introduce two parameters that evaluate both these properties. In this work, the metrics proposed by M. Vasile were adopted [15]:

$$M_{conv} = \frac{1}{N_p} \sum_{i=1}^{N_p} \min_{j \in M_p} 100 \left\| \frac{g_j - f_i}{g_j} \right\| \quad (4)$$

$$M_{spr} = \frac{1}{M_p} \sum_{j=1}^{M_p} \min_{i \in N_p} 100 \left\| \frac{f_i - g_j}{g_j} \right\| \quad (5)$$

Given the  $M_p$  solutions  $g_j$  used to describe the global Pareto front, and the  $N_p$  elements  $f_i$  in the Pareto front obtained from a given run of the optimization algorithm,  $M_{conv}$  is the sum over  $N_p$  of the distance of each element in the particular front considered from the closest element of the global front. This figure of merit clearly indicates how close the obtained front is to the global one. On the other hand,  $M_{spr}$  is the sum over all the elements in the global Pareto front of the distance of each element in the global front from the closest one in the front obtained for the considered run. This parameter measures how well the individuals of the obtained front cover the whole global front. Other metrics were proposed in the literature that try to evaluate these two performance (e.g. [16]) and future work will address a more extensive analysis which includes at least some of them.

##### B. Integral approach for comparison metrics

A Gaussian PDF is often assumed for the performance parameters, described by mean value and variance, computed by taking into account only those runs that have at least one

individual in the feasible region. But this simple approach has a significant methodological and practical fault that needs to be underlined, because the *a priori* hypothesis that the PDFs are Gaussian is usually far from true. As an example, both multi-objective performance indices,  $M_{conv}$  and  $M_{spr}$ , would be 0 at convergence, for  $N_{MAX} \rightarrow \infty$ , with a PDF represented by a Dirac function centered in zero. For  $N_{MAX} < \infty$  both metrics are strictly positive by definition, so that for very high values of  $N_{MAX}$  one expects a PDF more similar to an exponential or a  $\chi^2$  rather than to a Gaussian one. Only for relatively low values of  $N_{MAX}$  the values of  $M_{conv}$  and  $M_{spr}$  will be distributed on both sides of the most likely value.

The actual PDF shape depends on a) the considered problem, b) the algorithm and c) the value of  $N_{MAX}$ . Its shape is unknown and it will be shown to be multi-modal in the Section of results. In the absence of any actual knowledge about the true shape of the PDF, a practical and useful, yet correct and rigorous approach is based on extracting from the test results the success probability, which is the probability that the considered index is beyond a predefined threshold. As an example,  $P_S(M_{conv} < \theta_{conv})$  is the probability that the index  $M_{conv}$  achieves a value less than the threshold  $\theta_{conv}$ . These probabilities can be evaluated over a limited number of runs with far a better confidence than the PDF, so that it provides a more reliable merit function for optimization algorithm capabilities.

At this point, two indices of success,  $P_{FS, M_{conv}}$  and  $P_{FS, M_{spr}}$ , can be derived by combining constraint satisfaction probability and multi-objective requirements, that is:

$$\begin{aligned} P_{FS, M_{conv}} &= P_{FS} P_S(M_{conv} < \theta_{conv}) & (6) \\ P_{FS, M_{spr}} &= P_{FS} P_S(M_{spr} < \theta_{spr}) & (7) \end{aligned}$$

The first one is the product of the probability to find at least one solution in the feasible region,  $P_{FS}$ , times the probability that the index  $M_{conv}$  has a value less than the threshold  $\theta_{conv}$ . If one assumes that  $M_{conv} = \infty$  for those runs which are not able to find feasible solutions,  $P_{FS, M_{conv}}$  is equivalent to  $P_S(M_{conv} < \theta_{conv})$  computed on the basis of the whole set of runs.

### C. Critical aspects and practical solutions

In order to compute the two multi-objective metrics,  $M_{conv}$  and  $M_{spr}$ , the knowledge of the global front is required, either in analytic form or as a large set of global solutions. At the moment, such an information is not available, but it is possible to extract the best approximation of the global front from the whole set of available solutions, to be used as the reference global front.

Another important aspect that needs to be pointed out is that the statistical properties of the success indices, defined in the form described above, can be represented by means of a binomial PDF independently of the number of function evaluations, the problem formulation and the optimization algorithm. As a major consequence of this property, the test can be designed knowing *a priori* the relation between the

number of runs,  $N_{run}$ , and the error on the estimation of the success index [7]. As a drawback for multi-objective problems, the choice of the threshold is somehow arbitrary, inasmuch as it is not possible to define *a-priori* a value for an expected performance level, whereas for single-objective problems a reasonable threshold can often be determined, *e.g.* in terms of relevant values for the considered application.

A commonly adopted starting point for sizing the sample of a binomial distribution is to assume that both the normal approximation for the sample proportion  $p$  of successes (i.e.  $p \sim N\{\theta_p, \theta_p(1 - \theta_p)/n\}$ , where  $\theta_p$  is the unknown true proportion of successes) and the requirement that  $Pr[|p - \theta_p| \leq d_{err}|\theta_p]$  are at least equal to  $1 - \alpha_p$  [17]. This leads to expression  $N_{run} \geq \theta_p(1 - \theta_p)\chi_{(1), \alpha_p}^2/d_{err}^2$  that can be approximated conservatively with  $N_{run} \geq 0.25\chi_{(1), \alpha_p}^2/d_{err}^2$ , valid for  $\theta_p = 0.5$ . In the framework of this test campaign the number of runs for each test case was  $N_{run} = 200$ , so that one has  $d_{err} \geq (0.25\chi_{(1), \alpha_p}^2/N_{run})^{1/2}$ . For  $N_{run} = 200$ , with a 95% confidence level ( $\alpha_p = 0.05$ ), the measured success index is affected by an error as small as  $d_{err} \lesssim 0.05$ .

## V. RESULTS

In order to allow for a fair and easy comparison between the two optimization codes over the considered cases, code parameters were kept fixed, whenever possible. For the NSGA-II code, the following values were adopted: crossover probability,  $p_c = 0.9$ ; mutation probability,  $p_m = 1/dim$  (where  $dim$  is the dimension of the search space); distribution indices for crossover and mutation,  $\eta_c = 5$  and  $\eta_m = 5$ , respectively (values that should allow a good exploration of the search space). For the MOPED code the parameters were: number of constraint classes,  $N_{cl} = N_{ind}/10$ ; fitness coefficient,  $\alpha = 0.5$ ; sampling proportion,  $\tau = 1$ .

For both codes,  $N_{ind}$  and the maximum number of generations were set for each case on the basis of the expected degree of difficulty. All the results in terms of performance indices are listed in Table II. Each particular case will be detailed and commented separately in the next subsections.

### A. Case 1 (2-impulse transfer with Lambert formulation)

In this case, where a Lambert formulation is used, the optimisation algorithms were set with  $N_{ind} = 100$  and  $N_{genMAX} = 20$ , for a total of 2000 evaluations of the system model, with the following bounds on design parameters:  $t_W = x_1 \in [0, 10.8]$  h, and  $t_T = x_2 \in [0.03, 10.8]$  h.

Performance of the algorithms can be considered equivalent for the first test case. Constraint satisfaction and convergence to the front are reached by both codes:  $P_{FS} = 1$  and  $M_{conv}$  achieves small values, as reported in Tab. II. If on one side, a single run of the MOPED algorithm underperformed with respect to the prescribed threshold, the PDFs of  $M_{conv}$  (Fig. 2) show a slight advantage of the EDA algorithm over NSGA-II. On the other hand, both the algorithms are not able to spread the individuals over the whole front, as demonstrated by the high values of  $M_{spr}$  (Fig. 3), none of the algorithms being able to get below the prescribed threshold,  $\theta_{spr} = 1.25$ . In particular, MOPED is able to find solutions

TABLE II  
PERFORMANCE INDICES\*

	$\bar{M}_{conv}$	$\sigma^2(M_{conv})$	$\bar{M}_{spr}$	$\sigma^2(M_{spr})$	$P_{FS}$	$P_{S,M_{conv}}$	$P_{S,M_{spr}}$	$P_{FS,M_{conv}}$	$P_{FS,M_{spr}}$
<b>Case 1</b>									
MOPED	0.0945	0.0028	4.2851	0.8813	1	0.995	0	0.995	0
NSGA2	0.1347	0.0020	1.6122	0.0110	1	1	0	1	0
<b>Case 2</b>									
MOPED	27.2318	309.3810	5.7931	2.6710	1	0	0	0	0
NSGA2	3.9298	16.2396	3.4265	12.7370	1	0.08	0.285	0.08	0.285
<b>Case 3</b>									
MOPED	0.3662	0.0097	1.7345	1.8700	1	0.4850	0.495	0.485	0.495
NSGA2	52.8678	6996.9458	242.3426	47688.9646	0.605	0.0579	0	0.0350	0

\*For all the cases:  $N_{run} = 200$ . For 2-impulse transfers:  $\theta_{conv} = 0.35$ ,  $\theta_{spr} = 1.25$ ; for 3-impulse transfers:  $\theta_{conv} = 0.5$ ,  $\theta_{spr} = 1.5$

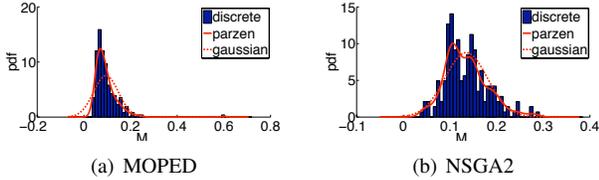


Fig. 2. Pdf of  $M_{conv}$  for Case 1

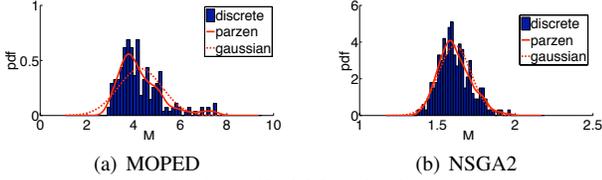


Fig. 3. Pdf of  $M_{spr}$  for Case 1

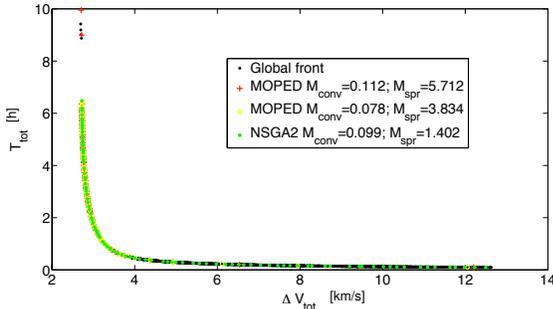


Fig. 4. Global front for Case 1 and examples of obtained front and related metric values.

belonging to the left-upper part of the front (red population in Fig. 4), thanks to its good exploring capabilities, but this results into a penalty in terms of  $M_{spr}$ , if this is made at the expenses of a worse representation of the bottom-right section of the front. NSGA-II performs better, on average, but none of the 200 runs was able to reach the separated portion of the front, induced by the existence of several local minima for the total  $\Delta V$ , corresponding to marginal fuel savings at the expenses of considerable higher total transfer time.

### B. Case 2 (3-impulse transfer with Lambert formulation)

For the three-impulse transfer the number of individuals was kept,  $N_{ind} = 100$ , but a higher number of generations was necessary,  $N_{genMAX} = 300$ , because of the increased difficulty of the problem. The bounds on the optimization variables are:  $t_W = x_1 \in [0, 1.62]$  h,  $t_{T,1} = x_2 \in [0.03, 21.54]$  h,  $r_2 = x_3 \in [7010, 105410]$  km,  $\Delta\theta = x_4 \in [0.01, 2\pi - 0.01]$  rad, and  $t_{T,2} = x_5 \in [0.03, 21.54]$  h.

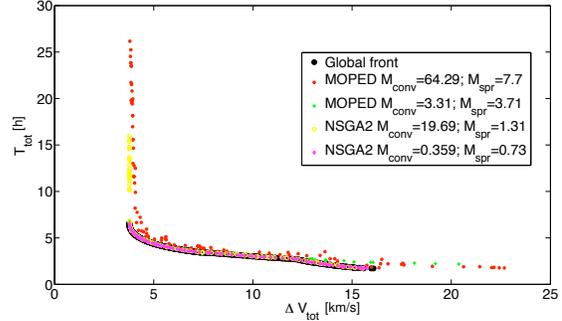


Fig. 5. Global front for Case 2 and examples of obtained front and related metric values.

In this case the performance of the two algorithms are quite different, NSGA-II appearing to be able to converge to the global front with greater accuracy. Although  $P_{FS,M_{conv}}$  is only 8% for NSGA-II, MOPED performs even worse, none of the runs achieving the prescribed threshold, and with an average value of  $M_{conv}$  almost 7 times higher. This situation is apparent from Fig. 5, where the global front and the fronts from two typical runs for each algorithm are reported. This behaviour is probably induced by the presence of a local front that traps the evolution of a considerable portion of the population, thus preventing it from converging to the actual Pareto front of the problem.

### C. Case 3 (2-impulse transfer with orbit propagation)

Case 3 deals with the same manoeuvre as Case 1, but because of the formulation in terms of orbit propagation, the resulting optimization problem proved to be significantly more complex, requiring a larger population ( $N_{ind} = 200$ ) and a higher number of generations ( $N_{genMAX} = 600$ ), for a total of 120000 evaluations of the system model. The tolerances on the constraints on the final orbit were set as  $\varepsilon_a = 40$  km,  $\varepsilon_e = 0.002$ ,  $\varepsilon_\theta = 0.01$  rad, and  $\varepsilon_\omega = 0.01$  rad. The optimization parameters were bounded as follows:  $x_1 \in [-0.2\pi, 14\pi]$  rad,  $x_2 \in [1.6, 4]$  km/s,  $x_3 \in [-0.2\pi, 0.2\pi]$  rad,  $x_4 \in [0.2\pi, 2\pi]$  rad,  $x_5 \in [0.004, 1.6]$  km/s, and  $x_6 \in [-0.6\pi, 0.6\pi]$  rad.

In the framework of the considered constrained problem, the MOPED algorithm outperforms the NSGA-II code. The exploring capabilities allow the EDA to better scan the search space, being able to always find feasible solutions. NSGA-II has a good capability of finding feasible solutions,  $P_{FS}$  being

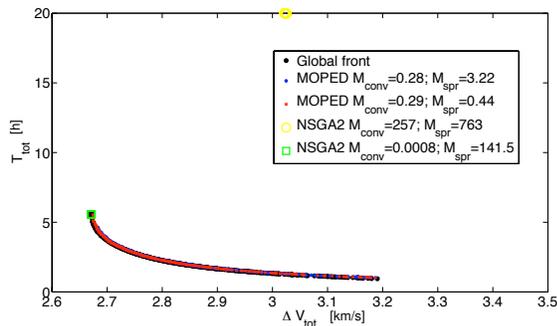


Fig. 6. Global front for Case 3 and examples of obtained front and related metric values.

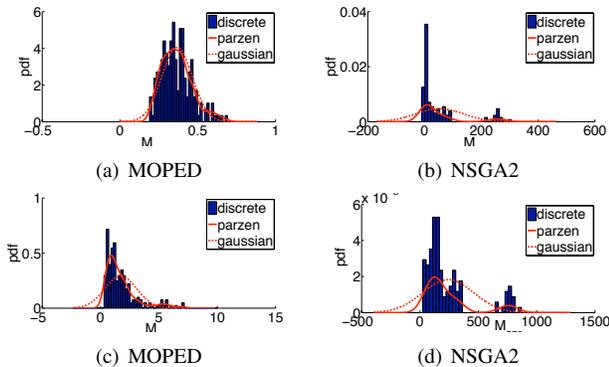


Fig. 7. Pdf of  $M_{conv}$  and  $M_{spr}$  for Case 3

more than 60%, but in the process the population is driven towards a relatively small portion of the feasible region, thus reducing the amount of available information for the search process. Figure 6 shows in yellow and green two “fronts” of feasible individuals obtained at the end of the optimisation process by the NSGA-II algorithm. The yellow one is a particularly bad run, which did not even converge to the front. The green one reaches exactly one of the edges of the front, but the code is then not capable of spreading the population along the front. In this latter case, although  $M_{conv}$  is close to 0,  $M_{spr}$  achieves again a high value. In this respect, the PDFs reported in Fig. 7 also demonstrate that MOPED algorithm performance are indubitably better, most of the runs achieving smaller values of both performance indices. At the same time, before drawing final conclusions, one should consider that the global front, derived according to the procedure outlined in Section IV.C, was made almost entirely by individuals taken from MOPED runs, thus giving it a clear advantage over the NSGA-II when metrics are evaluated.

## VI. CONCLUSIONS

The statistical properties of performance metrics that evaluate the capabilities of evolutionary algorithms of finding the correct representation of the Pareto front for multi-objective constrained optimisation problems were analyzed in the attempt of deriving reliable selection criteria between different methods. The considered metrics, applied to the evaluation of performance of two different Evolutionary Algorithms over three different test cases in the framework of the optimisation of impulsive orbit transfers, estimate the capability of finding feasible solutions, while converging to the Pareto front, spreading the population along it.

The study demonstrates that, in spite of a considerable computation effort (as many as 200 runs per test case were performed), it is not possible to rank the algorithms with sufficient confidence. The convergence index proved to be an effective measure for the capability of a method of converging onto a portion of the front, but the “spreadness” index proved to be less useful, if defined by means of a single merit function. The difficulty is related to intrinsic limits in the knowledge and representation of the reference solution (e.g. the “true” front) against which to compare each run. The need for improving both these aspects was outlined and it will be the subject of future research.

## REFERENCES

- [1] P.L. Di Lizia, G.M. Radice, D. Izzo, M. Vasile, “On the Solution of Interplanetary Trajectory Design Problems by Global Optimisation Methods” *Proc. International Workshop on Global Optimization GO 2005*, Almera (Spain), Sept. 2005, pp. 1–7.
- [2] M. Mitchell, *An introduction to genetic algorithms*, Cambridge (MA): MIT Press, 1998.
- [3] P. J. Werbos, H.G. Beyer, and H.P. Schwefel, “Evolution Strategies: A Comprehensive Introduction” *Natural Computing*, vol. 1, no. 1, pp. 3–52, Mar. 2002.
- [4] K.V. Price, R.M. Storn, and J.A. Lampinen, *Differential Evolution. A Practical Approach to Global Optimization*, Natural Computing Series, New York: Springer, 2005.
- [5] R. Biesbroek, “A Comparison of Differential Evolution Method with Genetic Algorithms for Orbit Optimisation”, IAC-06-c1.4.02, *Proc. 57<sup>th</sup> International Astronautical Congress*, Valencia, Spain, Sept. 2006.
- [6] M. Rosa Sentinella, “Comparison and Integrated Use of Differential Evolution and Genetic Algorithms for Space Trajectory Optimisation” *Proc. IEEE Congress on Evolutionary Computation (CEC) 2007*, Singapore, Sept. 2007.
- [7] M. Vasile, E. Minisci, and M. Locatelli, “On Testing Global Optimization Algorithms for Space Trajectory Design”, AIAA paper 2008-6277, *Proc. AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, Honolulu (HI), Aug. 2008.
- [8] E. Minisci, and G. Avanzini, “Comparative Study on the Application of Evolutionary Optimization Techniques to Orbit Transfer Maneuvers”, IAC-08-C1.2.10, *Proc. 59<sup>th</sup> International Astronautical Congress*, Glasgow (Scotland), Sept. 2008.
- [9] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II”, *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
- [10] M. Costa, E. Minisci, “MOPED: a Multi-Objective Parzen-based Estimation of Distribution algorithm”, *Proc. Second International Conference on Evolutionary Multi-Criterion Optimization, EMO 2003*, pp. 282–294, Faro (Portugal), Apr. 2003.
- [11] R.H. Battin, *An Introduction to the Mathematics and Methods of Astrodynamics*, Revised Ed., Reston (VA): AIAA Education Series, 1999 (Chapters 4,5).
- [12] G. Avanzini, E.A. Minisci, and D. Biamonti, “Minimum-Fuel/Minimum-Time Maneuvers of Formation Flying Satellites”, *Proc. AAS/AIAA Astrodynamics Specialist Conference*, AAS Paper 03-654, Big Sky (MT), Aug. 2003.
- [13] G. Avanzini, “A Simple Lambert Algorithm”, *Journal of Guidance, Control, and Dynamics*, Vol. 31, no. 6, pp. 1587–1594, Nov.–Dec. 2008.
- [14] N. Khan, D.E. Golberg, and M. Pelikan, “Multi-objective Bayesian Optimization Algorithm”, Technical Report IlliGAL 2002009, University of Illinois at Urbana-Champaign, 2002.
- [15] M. Vasile, “Hybrid Behavioural-Based Multiobjective Space Trajectory Optimization”, in *Multi-Objective Memetic Algorithms*, Springer Series of Studies in Computational Intelligence, to appear 2009.
- [16] E. Zitzler, K. Deb and L. Thiele, “Comparison of Multiobjective Evolutionary Algorithms: Empirical Results”, *Evolutionary Computation*, Vol. 8, No. 2, pp. 173–195.
- [17] C.J. Adcock, “Sample size determination: a review”, *The Statistician*, vol. 46, no. 2, pp. 261–283, Feb. 1997.