mathematics

Article

A New Hybrid Evolutionary Algorithm for the Treatment of Equality Constrained MOPs

Oliver Cuate [1,*], Antonin Ponsich [2], Lourdes Uribe [3], Saúl Zapotecas-Martínez [4], Adriana Lara [3] and Oliver Schütze [1,5]

[1] Department of Computer Science, Cinvestav-IPN, Mexico City 07360, Mexico; schuetze@cs.cinvestav.mx
[2] Metropolitan Autonomous University, Azcapotzalco Unit, Av. San Pablo No. 180, Col. Reynosa Tamaulipas, Azcapotzalco 02200, Mexico; aspo@azc.uam.mx
[3] Instituto Politécnico Nacional, Mexico City 07738, Mexico; lourdesur@esfm.ipn.mx (L.U.); adriana@esfm.ipn.mx (A.L.)
[4] Department of Applied Mathematics and Systems, Metropolitan Autonomous University, Cuajimalpa Unit (UAM-C), Vasco de Quiroga 4871, Santa Fe Cuajimalpa 05370, Mexico; szapotecas@correo.cua.uam.mx
[5] Dr. Rodolfo Quintero Ramirez Chair, Metropolitan Autonomous University, Cuajimalpa Unit (UAM-C), Vasco de Quiroga 4871, Santa Fe Cuajimalpa 05370, Mexico
* Correspondence: ocuate@computacion.cs.cinvestav.mx

Received: 27 November 2019; Accepted: 5 December 2019; Published: 18 December 2019

Abstract: Multi-objective evolutionary algorithms are widely used by researchers and practitioners to solve multi-objective optimization problems (MOPs), since they require minimal assumptions and are capable of computing a finite size approximation of the entire solution set in one run of the algorithm. So far, however, the adequate treatment of equality constraints has played a minor role. Equality constraints are particular since they typically reduce the dimension of the search space, which causes problems for stochastic search algorithms such as evolutionary strategies. In this paper, we show that multi-objective evolutionary algorithms hybridized with continuation-like techniques lead to fast and reliable numerical solvers. For this, we first propose three new problems with different characteristics that are indeed hard to solve by evolutionary algorithms. Next, we develop a variant of NSGA-II with a continuation method. We present numerical results on several equality-constrained MOPs to show that the resulting method is highly competitive to state-of-the-art evolutionary algorithms.

Keywords: multi-objective optimization; equality constraints; evolutionary algorithm; continuation method

1. Introduction

In many applications, one is faced with the problem that several objectives have to be optimized concurrently. One main characteristic of such multi-objective optimization problems (MOPs) is that their solutions sets typically form objects of dimension $k-1$, k being the number of objectives in the problem. Multi-objective evolutionary algorithms (MOEAs, e.g., [1,2]) have caught the interest of many researchers for the treatment of such problems as they have shown their efficiency both on academic and real-world problems. The population-based approach of MOEAs allows computing finite-size approximations of the entire solution set in one single run of the algorithm. Further, these methods are of global nature and require just minimal assumptions on the model (e.g., they can be work without gradient information).

Equality constraints represent an important class of constraints that naturally arise in many applications (e.g., [3–9]), and their importance will rise since decision making processes are getting more and more complex. Such constraints, however, still represent a challenge for state-of-the-art MOEAs ([10]).

In this work, we argue that a hybridization of a recently proposed Pareto Tracer (PT, [11]) with MOEAs leads to a new solver for continuous constrained MOPs that is fast and reliable. To this end, we propose and discuss here three equality constrained MOPs to complement the comparisons. Next, we propose ϵ-NSGA-II/PT that is a hybrid of a variant of the well-known and widely used NGSA-II and PT. PT is currently one of the fastest multi-objective continuation methods that allow us to trace solutions along the Pareto set/front, can be applied to higher dimensions, and reliably handles constraints. However, as all continuation methods, it is of local nature. We demonstrate the strength of the novel approach by showing results on selected benchmark problems and comparisons against some state-of-the-art MOEAs. A preliminary study of this work can be found in [12] which is restricted to the treatment of bi-objective problems, and where the discussion and comparison of the novel method is reduced.

This paper is structured as follows: in Section 2, we shortly recall some required background. In Section 3, we propose three new equality constrained MOPs that we will use for our comparisons. In Section 4, we propose the hybrid evolutionary algorithm ϵ-NSGA-II/PT. In Section 5, we present results on some benchmark problems including a comparison to the performance of related algorithms. In Section 6, we finally conclude our work and discuss possible further steps that can be made in this line of research.

2. Background and Related Work

2.1. Multi-Objective Optimization Problem (MOP)

In the sequel, we will consider continuous MOPs

$$\begin{aligned} &\min_{x\in\mathbb{R}^n} F(x) \\ &\text{s.t. } h_i(x) = 0 \; i = 1,\ldots,p \\ &\qquad g_j(x) \leq 0 \; j = 1,\ldots,m, \end{aligned} \tag{1}$$

where $F(x) = (f_1(x),\ldots,f_k(x))^T$ defines the objective functions. Q denotes the domain of (1), $Q := \{x \in \mathbb{R}^n \; : \; h_i(x) = 0 \quad \wedge \; g_j(x) \leq 0\}$. We say that $y \in \mathbb{R}^n$ is dominated by $x \in \mathbb{R}^n$ (in short $x \prec y$) if $f_i(x) \leq f_i(y)$, $i = 1,\ldots,k$, and it holds $f_j(x) < f_j(y)$ for a $j \in \{1,\ldots,k\}$. Else we say that y is non-dominated by x. $x \in Q$ is said to be (Pareto) optimal or simply optimal if there exists no $y \in Q$ with $y \prec x$. The Pareto set P_Q is the set of all optimal points, and its image $F(P_Q)$ is called the Pareto front. We have to assume that the gradients of all functions (objective and constraints) can at least be approximated in order to apply PT.

2.2. Related Work

The development of solution tools for the constrained MOP described in the previous subsection has generated a recent interest, mainly within the evolutionary computation community. This subsection provides a review of previous works related to this field, focusing on pure evolutionary techniques and on hybrid strategies as well.

Regarding MOEAs, two constraint handling mechanisms, classically used in the single-objective optimization framework, are adapted in [13] to MOEA/D [14]. The first one is stochastic ranking (SR, [15]), which accounts for the need of directing the search according to both feasibility and objective value. Therefore, when comparing two solutions and at least one of them is infeasible, the comparison criterion is the objective function value with probability p_f, while, with probability $1 - p_f$, individuals are distinguished through the overall constraint violation ϕ (where for any solution x, $\phi(x) = \sum_{i=1}^{p} |h_i(x)| + \sum_{j=1}^{p} \max\{0, g_j(x)\}$). On the other hand, the constraint-domination principle (CDP) is a multi-objective extension [16] of Deb's feasibility rules, where feasible solutions are compared according to dominance. The computational experiments, performed on the CTP-series [17] and CF-series [18], demonstrate that MOEA/D-CDP consistently outperforms MOEA/D-SR concerning hypervolume, inverted generational distance (IGD) and set coverage. It also performs reasonably well when compared with IDEA [19] and DMOEA-DD [20].

Besides, the Infeasibility Driven Evolutionary Algorithm (IDEA) [19] is a modification of the classical CDP-based NSGA-II [16] that enforces the participation of infeasible solutions through a parameter α, which represents the fraction of the current population allocated for those solutions. During the selection step, the pool combining parent and offspring populations is first divided into two (feasible and infeasible) sets. Then, non-dominated ranking is applied to both these sets, considering a function of the constraint violations as an additional objective for the infeasible set. Then, if N is the population size, the $\alpha \cdot N$ infeasible solutions with the best-ranking values, as well as the $(1 - \alpha) \cdot N$ feasible solutions with the best ranks, are included into the next population. Note that the constraint violation function used as an additional objective for infeasible solutions is computed as the sum of the ranking of the solutions, sorted in increasing order of the magnitude of constraint violation for each constraint (instead of the number of violated constraints formerly used [21]). The experimental results conducted over some test functions of the CTP-series [17] demonstrate that IDEA consistently outperforms CDP-NSGA-II.

Another adaptation of MOEA/D for handling constrained MOPs is introduced in [22] where, for each solution, a modified function for the overall constraint violation accounts for the number of active constraints, in addition to the simple constraint violation. The mean value of this modified function over the population is weighted by the ratio of feasible individuals in the population, in order to produce a threshold on the allowed amount of constraint violation. Solutions that are within this threshold are considered as feasible and compared in terms of their objective values. Furthermore, a gradient-based local search is periodically invoked in order to repair infeasible solutions. The resulting algorithm, tested on some of the CTP-series test functions [17], performs similarly or better than NSGA-II with CDP.

In [23], the ϵ-constraint technique originally developed for single-objective optimization is extended for the solutions of MOPs. This strategy, proposed in [24], consists in relaxing the tolerance level on constraints up to a value ϵ. Thus, when two solutions have an overall constraint violation lower than ϵ, they are both considered as feasible and compared in terms of their objective value. In [24], the value of ϵ is monotonically decreased according to a polynomial function, until some generation T_c. From then on, ϵ is set to 0 in order to narrow the search on the feasible space. In [23], the authors allow increasing the ϵ level when the ratio of feasible solutions is greater than a threshold value, to promote exploration. This strategy, embedded in MOEA/D, is compared with MOEA/D-CDP [13] and with the original ϵ-constraint mechanism (decreasing ϵ pattern), over a set of nine constrained MOPs introduced earlier by the same authors [25]. Furthermore, this strategy is also compared with classical MOEAs (either dominance or decomposition-based) in a later work [26], over the CTP [17] and CF series [18], using IGD as a performance indicator. In any case, the MOEA/D-IEpsilon algorithm outperforms all its contenders, except IDEA [19], which obtains similar performance levels.

More recently, the same authors developed a two-stage (Push and Pull Search, PPS) procedure, which first solves the unconstrained MOP (solutions are pushed towards the unconstrained Pareto front) and, in a second stage, include constraints to modify the first (unconstrained) approximation and identify the constrained Pareto front (solutions are pulled from infeasible regions towards the feasible space). The switching criterion between both phases is based on no-evolution of the identified ideal and Nadir points. During the "push stage", the canonical MOEA/D (with the Tchebycheff scalarizing function) is employed, while a modified ϵ-constraint technique is applied in the second one (still with MOEA/D) to find feasible solutions. A specific decreasing scheme for the ϵ level is adopted, where exponential or polynomial (as in [24]) decrease can be used, depending on the feasibility ratio. Tested over a 14 functions benchmark earlier proposed by the same authors [25], the resulting PPS-MOEA/D outperforms some classical MOEAs quoted in this section but requires tuning many parameters.

As the last example of MOEA-based solution procedures, an innovative idea is introduced in [27], where MOEA/D is modified in such a way that two solutions are assigned to each weight vector. The aim is having one individual on each side of the feasibility boundary (one feasible and one infeasible), in order to focus the search on this region where the Pareto-optimal solutions might lie. The consequences of this working mode are: (i) the doubled size of the neighborhood of each weight vector for offspring generation (since each neighboring weight vector has two associated solutions); (ii) for solution replacement, the created offspring is now compared to two individuals, in terms of both the scalarizing function used within MOEA/D and the overall constraint violation. In this bi-objective space, dominance is used to select two surviving individuals among the three contenders. If the three are non-dominated, that with the larger constraint violation is discarded, while if one solution dominates the two others, the former is the only one to survive. The algorithm is successfully compared with MOEA/D-CDP over several functions of the C-DTLZ test suite [28]. However, for CTP and CF series [17,18], this Dual-grid MOEA/D is outperformed by IDEA [19] or MOEA/D-IEpsilon [26].

Besides, as for single-objective optimization, hybrid strategies have been adapted for solving MOPs in recent years. In general, these so-called memetic algorithms combine a global search engine (a MOEA) with some local search technique based on exact algorithms. In this framework, proposals differ one from another according to:

- The kind of local search technique used, which may be gradient-based (quasi-Newton in [29,30], or sequential quadratic programming in [31]) or direct search for nonlinear problems (Nelder and Mead's algorithm in [32]).
- The problem reformulation on which local search is applied, which may be based on ϵ-constraint [33] or a scalarization of the MOP [31].
- The hybridization scheme, which can consists on seeding the initial population of the MOEA [34], interleaving global and local search steps by applying local search to some selected individuals of the population [33] or periodically (every t generations) [35], or using the non-dominated solutions obtained by the exact algorithm to reconstruct the whole Pareto front [36].

However, to the best of our knowledge, most of these hybrid strategies were applied to unconstrained MOPs through classical test suites (ZDT, DTLZ, WFG) and there is almost no proposal for dealing with constraints, particularly equality constraints.

There already exist some strategies for deal with equality constraints: for instance, if all equalities are linear, one can use orthogonal projections in order to obtain feasible points near to a given candidate solution ([8]). For vehicle routing problems, several repair mechanisms can be found in [37–39]), and in [4,40] such mechanisms can be found for portfolio selection problems. Finally, [41] proposes a repair mechanism that makes use of first-order Taylor approximations of the constraints.

2.3. Test Suites for Constrained MOPs

Benchmark problems are important to assess the performance of solution techniques. In literature, some of such test suites can be found for the case of constrained MOPs. Most of these problems, however, contain only box-constraints. Established test suites whose domains are more complex are the CTP problems ([17]) and the CF problems (see [18]). More recently, the C-DTLZ ([28]) and the LIRCMOP problems ([25]) have been proposed. Remarkably, not one of these test problems contains an equality constraint. Thus, the common operating mode proposed by most authors to handle equalities (i.e., dividing the constraint into two inequalities) has never been seriously tested. Indeed, this methodology does not work in practice since it is very unlikely that a solution can respect both inequalities. As a consequence, the first feasible solution identified attracts the rest of the population, causing drastic diversity losses. This is why the specific treatment of equality constraints represents a relevant issue on its own, as shown by the present study.

In [42], the CZDT functions are proposed that contain equality constraints and that are derived from the well-known ZDT functions ([43]). We will use this test suite for our comparisons. The particularity of this suite, however, is that the Pareto sets of the constrained problems are identical to the Pareto sets of the respective unconstrained problems. To further test the ability of MOEAs to handle constraints, we will in this paper propose three new equality constrained MOPs where the inclusion of the equality constraint(s) has an influence on the location of the Pareto sets.

2.4. Pareto Tracer (PT)

PT is a continuation method that is capable of performing a movement along the set of Karush–Kuhn–Tucker (KKT) points of a given MOP starting from an initial KKT point [11]. The algorithm can handle both equality and inequality constraints and is applicable in principle to any number of objectives. The main steps of PT for equality constrained MOPs are for convenience of the reader briefly described in Appendix A.

3. Proposed Test Problems

Here we propose two bi-objective test problems, Eq1-ZDT1 and Eq2-ZDT1, and one three-objective problem, Eq-Quad. All problems are scalable in the number of decision variables, and for all problems the inclusion of the equality constraint(s) has an influence on the location of the Pareto set.

3.1. Eq1-ZDT1

The original ZDT1 is a bi-objective problem with box constraints that can be defined for an arbitrary number n of decision variables. Meanwhile, the proposed Eq1-ZDT1 problem is stated as follows

$$\begin{aligned} f_1(x) = &\quad x_1 \\ f_2(x) = &\quad g(x)\left(2 - \sqrt{\frac{f_1(x)}{g(x)}}\right) \end{aligned} \tag{2}$$

$$\text{s.t.} \quad h(x) = (x_1 - 0.5)^2 + (x_2 - 0.4)^2 - 0.25 = 0 \tag{3}$$

where

$$g(x) = 1 + \frac{9}{n-1} \sum_{i=2}^{n} x_i^2 \tag{4}$$

As we can see, the Eq1-ZDT1 (2) is also a scalable bi-objective problem in the number of variables, which changes the box constraints by an equality constraint (with the implicit inequality constraint that $x_1 \geq 0$ so that f_2 is defined). The constraint of this problem (3) defines a kind of "hyper-cylinder", where the variables x_1 and x_2 are placed on a circle, while the remaining variables $x_i,\ i = 3, \ldots, n$ can take any value (see Figure 1).

In the following, we will provide the Pareto set for Eq1-ZDT1. For this, we need the set $P_h \subset Q$ defined as

$$\begin{aligned} P_h := \Big\{ & x \in \mathbb{R}^n : (x_1 - 0.5)^2 + (x_2 - 0.4)^2 = 0.25, \\ & x_i = 0,\ i = 3, \ldots, n. \} \end{aligned} \tag{5}$$

Theorem 1. *Let P_h be defined as in (5) and $n = 30$. Then the subset $\mathcal{P}_{Eq1} \subset P_h$ given by*

$$\begin{aligned} &\mathcal{P}_{Eq1} = \{ x \in \mathbb{R}^n : x_1 \in [0, \gamma], \\ &x_2 = 0.4 - \sqrt{0.25 - (x_1 - 0.5)^2},\ x_i = 0,\ i = 3, \ldots, n \}. \end{aligned} \tag{6}$$

where $\gamma \approx 0.977336$ is the Pareto set of Eq1-ZDT1.

Proof.

(a) First, we prove that $\nexists\, y \in Q \setminus P_h$ such that $y \prec x$, where $x \in P_h$.

Suppose that $\exists\, y \in \mathbb{R}^n \setminus P_h$ such that $h(x) = 0$ and $y \prec x, \forall x \in P_h$. First, let

$$x := (y_1, y_2, 0, \ldots, 0)^T, \tag{7}$$

with $(y_1 - 0.5)^2 + (y_2 - 0.4)^2 = 0.25$. Then, let $\Delta := (\Delta_1, \ldots, \Delta_n)^T \in \mathbb{R}^n \setminus \emptyset$ with $\Delta_1 = \Delta_2 = 0$; as $y \in \mathbb{R}^n \setminus P_h$ we can choose y as follows:

$$y := x + \Delta = (y_1, y_2, \Delta_3, \ldots, \Delta_n)^T, \tag{8}$$

Now, from (2) note that

1. For the first objective we have that $f_1(y) = f_1(x) = y_1$.
2. For the second objective we have that

$$\begin{aligned} g(x) &= 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i^2 \\ &= 1 + \frac{9}{n-1}\left(y_2^2 + \sum_{i=3}^{n} 0\right) \\ &= 1 + \frac{9}{n-1} y_2^2, \\ g(y) &= 1 + \frac{9}{n-1}\sum_{i=2}^{n} y_i^2 \\ &= 1 + \frac{9}{n-1}\left(y_2^2 + \sum_{i=3}^{n} \Delta_i^2\right), \\ \Rightarrow g(x) &< g(y). \end{aligned}$$

Then,

$$\begin{aligned} \frac{y_1}{g(x)} > \frac{y_1}{g(y)} &\Rightarrow \sqrt{\frac{y_1}{g(x)}} > \sqrt{\frac{y_1}{g(y)}} \\ &\Rightarrow -\sqrt{\frac{y_1}{g(x)}} < -\sqrt{\frac{y_1}{g(y)}} \\ &\Rightarrow 2 - \sqrt{\frac{f_1(x)}{g(x)}} < 2 - \sqrt{\frac{f_1(y)}{g(y)}}. \end{aligned}$$

Finally,

$$\begin{gathered} g(x) < g(y) \Rightarrow \\ g(x)\left(2 - \sqrt{\frac{f_1(x)}{g(x)}}\right) < g(y)\left(2 - \sqrt{\frac{f_1(y)}{g(y)}}\right) \\ \Rightarrow f_2(x) < f_2(y), \end{gathered}$$

which contradicts the hypothesis. Thus $\nexists\, y \in P_D \setminus P_h \,|\, x \prec y$ with $x \in P_h$.

(b) Now we show that the points $\mathcal{P}_{Eq1}$ are not dominated by each other and they dominate all the points in the set $P_h \setminus \mathcal{P}_{Eq1}$.

Let $x, x' \in P_h$ such that $x_1 = x'_1$ and $|x_2| < |x'_2|$. Notice that we can express x_2 in terms of x_1 as

$$x_2 = 0.4 \pm \sqrt{0.25 - (x_1 - 0.5)^2}, \tag{9}$$

and it is clear that the points of the form $(x_1, 0.4 + \sqrt{0.25 - (x_1 - 0.5)^2})$ are dominated by the points $(x_1, 0.4 - \sqrt{0.25 - (x_1 - 0.5)^2})$ (that is, the inferior half of the circle), then $g(x) < g(x') \Rightarrow f_2(x) < f_2(x')$.

Now, we can write $f_2(x)$ with $x \in P_{h_d}$ in terms of x_1 as

$$f_2(x_1) = \left(\frac{(n-1) + 9C^2(x_1)}{n-1}\right)\left(2 - \frac{x_1}{(n-1) + 9C^2(x_1)}\right). \tag{10}$$

where, $C(x_1) = 0.4 - \sqrt{0.25 - (x_1 - 0.5)^2}$.

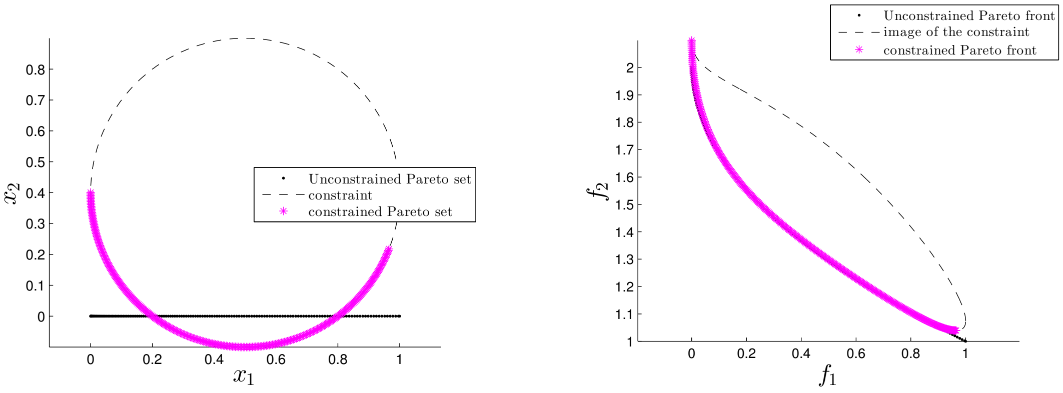
Computing the derivative of Equation (10) we have

$$f'_2(x_1) = -\frac{(5(n-1) + 180x_1 - 90)\sqrt{0.25 - (x_1 - 0.5)^2} - 72x_1 + 36}{5(n-1)\sqrt{0.25 - (x - 0.5)^2}}. \tag{11}$$

The derivative of f_2 has only one root at $\gamma = x_1^* \approx 0.977336$. Also, note that for a point a we have that if $a \in [0, \gamma]$, then $f'_2(a) < 0$. On the other hand, if $a \in [\gamma, 1]$, then $f'_2(a) > 0$. Hence, $f_2(x)$ is monotonically decreasing, and consequently, points in $\mathcal{P}_{Eq1}$ are not dominated by each other.

Finally, by (a) and (b) we have that $\mathcal{P}_{Eq1}$ is the Pareto set of Eq1-ZDT1. □

To obtain γ for the formulation of the Pareto set we needed to consider f'_2 which depends on n. The proof is analog for other values of n with changing value of γ. Some of these values can be found in Table A1 of Appendix D.



(**a**) (**b**)

Figure 1. Pareto set and front of the Eq1-ZDT1 with $n = 30$. (**a**) Projection of the Pareto set onto the x_1x_2-plane; (**b**) Pareto front.

3.2. Eq2-ZDT1

Via adding box constraints to Eq1-ZDT1, we can define the Eq2-ZDT1 problem as follows

$$\begin{aligned} f_1(x) = &\quad x_1 \\ f_2(x) = &\quad g(x)\left(2-\sqrt{\frac{f_1(x)}{g(x)}}\right) \end{aligned} \tag{12}$$

$$\text{s.t.} \qquad h(x) = 0 \tag{13}$$

$$0 \leq x_i \leq 1,\ i = 1,\ldots,n, \tag{14}$$

where $h(x)$ and $g(x)$ are defined as in (3) and (4), respectively.

Next, we will provide the analytical Pareto set for Eq2-ZDT1.

Theorem 2. *For $n = 30$, $x \in \mathbb{R}^n$, the Pareto set for the Eq2-ZDT1 problem (see Figure 2) is given by*

$$\begin{aligned} \mathcal{P}_{Eq2} := \{x \in \mathbb{R}^n : x_1 \in \mathcal{I}_1 \cup \mathcal{I}_2 \cup \mathcal{I}_3,\ x_2 = I(x_1), \\ x_i = 0,\ i = 3,\ldots,n.\} \end{aligned} \tag{15}$$

where $\mathcal{I}_1 := [0, 0.2]$, $\mathcal{I}_2 := [\eta, 0.8)$, $\mathcal{I}_3 := [0.8, \gamma]$, $\eta \approx 6.700214$, and

$$I(x_1) := \begin{cases} 0.4 - \sqrt{0.25 - (x_1 - 0.5)^2}, & x_1 \in \mathcal{I}_1 \cup \mathcal{I}_3 \\ 0.4 + \sqrt{0.25 - (x_1 - 0.5)^2}, & x_1 \in \mathcal{I}_2 \end{cases} \tag{16}$$

Proof.

(a) Let P_h be defined as in (5), and P_{Eq1} as in (6) and then first part of this proof is analogs the previous analysis for Eq1-ZDT1.

(b) As second step, we need to remove all the points in P_{Eq1} that do not satisfy the box constraints. In particular, as $x_i = 0,\ i = 3,\ldots,n$, we focus on x_1 and x_2. For $x_1, x_2 \in P_h$, we have that $x_1 \in [0, 1]$ and $x_2 \in [-0.1, 0.4]$, i.e., some values of x_2 do not satisfy the lower bound.

We can express x_1 as follows:

$$x_1 = 0.5 + \sqrt{0.25 - (x_2 - 0.4)^2}, \tag{17}$$

thus, for $x_2 = 0$ we can find the values of x_1 that define $\mathcal{I}_1$ and $\mathcal{I}_3$ via:

$$x_1 = 0.5 \pm 0.3 \Rightarrow \mathcal{I}_1 = [0, 0.2]\ \mathcal{I}_3 = [0.8, \gamma]$$

After removing the non-feasible points from P_{Eq1} we have a gap in Pareto set/front. Now, notice that, some points $x \in P_h : x_2 = 0.4 + \sqrt{0.25 - (x_1 - 0.5)^2}$ (that is, $x \notin P_{Eq1}$), could be within the gap. That is, we have to find the values of $x \in P_h \setminus P_{Eq1}$ such that $f_2(x_1) \in [f(0.8), f(0.2)]$.

For this we consider:

$$\bar{f}_2(x_1) = \left(\frac{(n-1) + 9C_2^2(x_1)}{n-1}\right)\left(2 - \frac{x_1}{(n-1) + 9C_2^2(x_1)}\right). \tag{18}$$

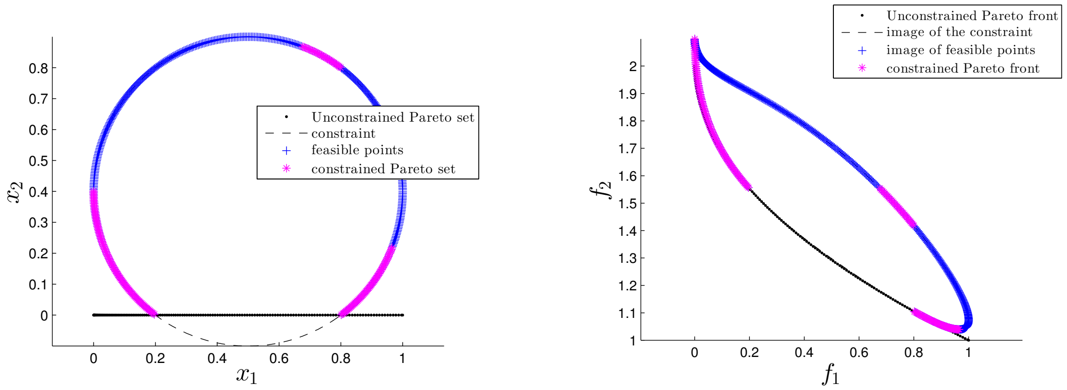
where, $C_2(x_1) = 0.4 + \sqrt{0.25 - (x_1 - 0.5)^2}$.

Notice that $[f_2(0.8), f_2(0.2)] \subset [\bar{f}_2(0.8), \bar{f}_2(0.2)]$ and $\bar{f}_2$ is a continuous function, then for the intermediate value theorem $\exists\, x_{1,0.8}, x_{1,0.2}$ such that $\bar{f}_2(x_{1,0.2}) = f_2(0.2)$ and $\bar{f}_2(x_{1,0.8}) = f_2(0.8)$, respectively.

For $n = 30$, such values are $x_{1,0.8} \approx 0.9773356$ and $x_{1,0.2} \approx 0.670021$. Then $\mathcal{I}_2 = [\eta, 0.8)$, with $\eta = x_{1,0.2}$, as $x_{1,0.8} > 0.8$ and $f_2(0.8) < \bar{f}_2(0.8)$.

Finally, by (a) and (b) we have that $\mathcal{P}_{Eq2}$ is the Pareto set of Eq2-ZDT2. □

As we can observe, the values of γ and η depend on n (see Theorems 1 and 2). We refer to the Appendix D for these values for other dimensions of the decision variable space. See Figure 2 for Pareto set and front of Eq2-ZDT2.



(**a**) (**b**)

Figure 2. Pareto set and front of the Eq2-ZDT1 with $n = 30$. (**a**) Projection of the Pareto set onto the x_1x_2-plane; (**b**) Pareto front.

3.3. Eq-Quad

Finally, we propose a modification of the problem taken from [11] which has three quadratic objectives and two equality constraints:

$$f_j(x) = \|x - a^{(j)}\|_2^2, \quad j = 1, \ldots, k,$$

where $x \in \mathbb{R}^n$, $k = 3$, and $a^{(1)} = (1, -1.4, -0.4)^T$, $a^{(2)} = (-1.4, 1, -0.4)^T$, and $a^{(3)} = (0.4, 0.4, 0.8)^T$. subject to

$$\begin{aligned} h_1(x) &= r^2 - x_3^2 - \left(R - \sqrt{x_1^2 + x_2^2}\right)^2 = 0 \\ h_2(x) &= x_1 + x_2 - x_3 = 0 \\ & -1.5 \le x_1 \le 1 \\ & -1.5 \le x_2 \le 1 \\ & \quad 0 \le x_3 \le 1. \end{aligned}$$

Figure 3 shows the constraints and the Pareto set for $n = 3$. As it can be seen, the Pareto set consists of two connected components that can be both expressed by curves (and which are hence 1-dimensional).

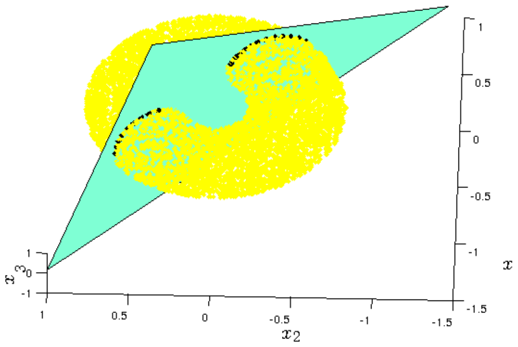


Figure 3. Constrains and feasible region for the torus problem.

4. Proposed Algorithm (ϵ-NSGA-II/PT)

This section is devoted to the description of the hybrid algorithm proposed in this work. As abovementioned, the Pareto Tracer is a continuation strategy that is able to efficiently perform moves along the Pareto front of a MOP. However, this reconstruction process is carried out locally, which involves that PT must be provided with a reduced set of relevant approximated solutions, i.e., with a reasonably good convergence and dispersion over the front. Therefore, a multi-objective evolutionary algorithm, based on a modified implementation of NSGA-II [16], is developed here as the first stage of the hybrid algorithm, in the aim of producing this first set of promising solutions. This latter subsequently serves as an input of PT, which refines it to produce the final approximation of the Pareto front/set. The two stages of the resulting ϵ-NSGA-II/PT are presented in detail in what follows.

4.1. First Stage: Rough Approximation via ϵ-NSGA-II

The first stage of the hybrid algorithm presented in this work consists in determining a rough approximation of the Pareto front via a MOEA. Each approximated solution will subsequently feed the Pareto Tracer, which generates local reconstructions of the real Pareto front and combines them to finally obtain the whole Pareto set. Therefore, the MOEA used in the first step should meet the following characteristics:

- the number of solutions in the roughly approximated set is small, in order to reduce the computational burden of the local search (PT). Indeed, in case of a completely connected front, one single approximated solution might allow us to build the entire Pareto front.
- the MOEA should promote diversity, since the rough approximation produced should cover all the extent of the Pareto front and identify all the different components, in case of a disconnected front.
- the MOEA must be able to handle equality constraints. As mentioned before, a severely constrained problem might cause diversity issues that should be overcome by the MOEA.

Note that the two first features are conflicting, since the number of elements of the approximated front should be small enough to avoid unnecessary computations. Nonetheless, there must be enough approximated solutions to ensure the identification of all the components, in case of disconnected Pareto front. To deal with 2 or 3 objective problems such as those treated in this work, we observed that 20 points in the rough approximation represents a good trade-off between these two requirements. This means that the MOEA should work either with small populations or maintaining a small external archive.

Regarding the search engine and diversity preservation issue, two classical dominance and decomposition-based algorithms (NSGA-II and MOEA/D, respectively) were initially investigated. However, preliminary computations led to the observations that, first, the use of small populations deteriorated the performance of MOEA/D and, even with larger populations (and a small archive to be returned), diversity remained an issue. Therefore, NSGA-II was used in this study as baseline algorithm.

In the proposed implementation, NSGA-II does not necessarily use small populations, but the final population is reduced to 20 at the end of the search (according to crowding distance). Regarding constraint handling, the constraint-domination principle CDP generally used within NSGA-II does not obtain good results on the equality constrained problems tackled here. Indeed, any technique establishing the superiority of feasible solutions over infeasible ones will face severe diversity issues, due to a premature convergence towards the first feasible solution found.

As a simple way to modify this operating mode, the ϵ-constraint strategy proposed in [24] is adopted here. Note that, despite this method was previously integrated within MOEA/D in [23], to the best of our knowledge, there is no reference to NSGA-II working with ϵ-constraint for solving constrained MOPs. As explained earlier, the ϵ-constraint strategy relaxes the tolerance level on constraint violation up to a value ϵ, i.e., a solution x with an overall constraint violation $\phi(x) \leq \epsilon$ is viewed as feasible and, thus, evaluated and compared in terms of its objective value. Then, ϵ is smoothly reduced according to a specific decreasing scheme. This technique allows solutions with a small (but positive) overall constraint violation to be compared with feasible individuals in terms of convergence (through dominance-based sorting) and spread uniformity (thanks to the crowding distance). Accordingly, the following operator is used to compare two solutions x and y:

$$x \prec_\epsilon y \iff \begin{cases} x \prec y & \text{if } \phi(x), \phi(y) \leq \epsilon \\ x \prec y & \text{if } \phi(x) = \phi(y) \\ \phi(x) < \phi(y) & \text{otherwise,} \end{cases} \tag{19}$$

where

$$\phi(x) = \sum_{j=1}^{m} \max\left\{g_j(x), 0\right\} + \sum_{i=1}^{p} |h_k(x)| \tag{20}$$

denotes the constraint violation. At the beginning of the run, the first value of ϵ, denoted as $\epsilon(0)$, is set in such a way that at least some solutions are "ϵ-feasible". More precisely, $\epsilon(0)$ is the overall constraint volation of individual x^θ, which is the θ-th solution in the first population sorted in descreasing order of ϕ: $\epsilon(0) = \phi(x^\theta)$. For subsequent generations, the ϵ decreasing schedule originally proposed in [24] is adopted here: ϵ is an exponentially decreasing function of the generation number, until a critical generation T_c is reached. From then on, ϵ is set to ϵ_{min}:

$$\epsilon = \begin{cases} \max(\epsilon_{min}, \epsilon(0)\,(1 - t/T_c)^{cp}) & \text{if } 0 < t < T_c \\ \epsilon_{min} & \text{if } t \geq T_c, \end{cases} \tag{21}$$

where t is the generation number, cp is a parameter controlling the decreasing speed and $\epsilon(0)$ is the constraint relaxation level at the first generation. Note that other ϵ decreasing schedules, such as that embedded within MOEAD/D-IEpsilon [23], have been tested, but better results were obtained using Equation (21).

Finally, in order to promote diversity, another parameter is introduced to bias the parent selection operator, which is based on a tournament implementing the ϵ-constraint strategy described in Equation (19). However, as suggested in [44], the result of the tournament is respected only with probability p_f; else (i.e., with probability $1 - p_f$), the winner is randomly chosen. Empirical results proved that this small modification sometimes allows significant improvements in the algorithm performance. The whole process is shortly described in Algorithm 1.

4.2. Second Stage: Refinement via PT

The main task on this stage is to process the resulting archive P provided by ϵ-NSGA-II. The main challenge here is to avoid unnecessary effort, e.g., via computing non-optimal KKT points along local Pareto fronts that are already dominated by previously computed solutions. While this is relatively easy for $k = 2$ objectives (see [12]), this task becomes more complicated with increasing k. The following procedure works for general k.

Before PT can be executed, the following post-processing has to be done on P:

1. Let τ be the desired minimal distance between two solutions in objective space. In this first step, go over P and eliminate elements that are too close to each other (if needed). This leads to the new archive $\tilde{P}$.
2. Apply the Newton method (A11) to all elements of $\tilde{P}$. Remove all dominated solutions, and elements that are too close to each other as in the first step. This leads to the archive $\bar{P}$.
3. To obtain a "global picture" of the part of the Pareto front that will be computed by PT, construct a partition of a potentially interesting subset S of the image space into a set of hyper-cubes (or k-dimensional boxes) with radius $\approx \tau$. This partition can easily be constructed via using a binary tree whose root represents S (see [45] for details, where, however, the partition is used in decision variable space). S is a box that is constructed out of $\bar{P}$ as follows: denote by m_i and M_i the minimal and maximal value of the i-th objective value of all elements in $\bar{P}$, respectively. Then the i-th element of the center of S is set to $(m_i + M_i)/2$ and its i-th element of the radius to $(M_i - m_i)/2$. In the computations, we will only allow a storing one candidate solution within each of these boxes in the archive A to guarantee a spread of the solutions.

Algorithm 1 ϵ-NGSA-II

```
$P \leftarrow pop\_init()$
Evaluate each individual $x^i \in P$ to obtain $F(x^i)$ and $\phi(x^i)$
Compute $\epsilon(0)$ and set $\epsilon = \epsilon(0)$
for $t \leftarrow 1$ to $MaxGen$ do
    $P' \leftarrow crossover(P)$        ▷ Parent selection through tournament and $\epsilon$-constraint
    $P'' \leftarrow mutation(P')$
    $Q \leftarrow P \cup P''$
    $Q^F \leftarrow Feasible(Q, \epsilon)$, $Q^I = Infeasible(Q, \epsilon)$
    $Q^F \leftarrow FastNonDominatedSorting(Q^F)$
    $Q^I = SortConstraintViolation(Q^I)$
    Fill $P$ with $Q^F$, using crowding distance if necessary
    if $|Q^F| < PopSize$ then
        Complete $P$ with $Q^I$
    end if
    Update $\epsilon$ through Equation (21)
end for
$P \leftarrow reduce(P)$        ▷ return 20 solutions
Return $P$ and $F(P)$
```

Then, in the first step the element $p^{(1)} \in \bar{P}$ is chosen as the starting point for PT, where $f_1(p^{(1)}) = m_1$. An external archive A will be created that will be the reference for PT and that will be the set of solutions that will be returned after the application of ϵ-NSGA-II/PT. At the beginning, it is $A := \{p^{(1)}\}$. In parallel, a box collection C will be created that contains all the (unique) boxes out of the above partition that contain all the elements of A. In the first step, C will set to the box that contains $p^{(1)}$. The application of PT leads to a sequence of candidate solutions x_i, $i = 1, \ldots, s$. For each x_i the following steps are performed:

1. if x_i is dominated by any element of A, then PT the current application of PT is stopped.
2. else, it is checked if the unique box that contains x_i is already contained in C. If this is not the case, add this box to C and add x_i to A. Else, decline x_i and proceed with x_{i+1}.

After this, take the element from $\bar{P}\backslash\{p^{(1)}\}$ with the smallest value of f_2 as the starting point for PT and proceed as above. Proceed in this way, sorting cyclic according to each objective, until all elements $p \in \bar{P}$ have been chosen as starting points for PT.

5. Numerical Results

Here we present some numerical results that compare the behavior of a well-known mathematical programming technique and some state-of-the-art MOEAs in the selected test problems. As test functions we have chosen to take the three test problems proposed above, Eq1-Quad from [11], the CZDT test suite as well as a modification of a problem from Das and Dennis problem (D&D) stated in [11]. First, we test the normal boundary intersection (NBI) method in the selected suite. With this experiment we want to show why the use of MOEAs is convenient, even more the need for a special hybrid algorithm which is capable of solving these type of MOPs. The second experiment is the comparison between state-of-the-art MOEAs against the proposed ϵ-NSGA-II/PT. Here, a point x is considered to be feasible if $\|h(x)\| < \varepsilon$, where we have taken $\varepsilon = 1e - 04$, which is common in evolutionary computation.

5.1. Performance Assessment

The multi-objective solvers were evaluated by adopting two performance indicators taken from the literature.

Δ_p indicator

The Δ_p indicator [46,47] can be viewed as an averaged Hausdorff distance between an approximation set and the real Pareto front of a MOP. This indicator is defined by slight modifications of the indicators Generational Distance (GD) [48] and Inverted Generational Distance (IGD) [49]. Formally, the Δ_p indicator can be written as follows.

Let $P = \{\vec{x}^1, \ldots, \vec{x}^{|P|}\}$ an approximation and $R = \{\vec{r}^1, \ldots, \vec{r}^{|R|}\}$ be a discretization of the real PF of a MOP, then

$$\Delta_p(P, R) = \max\{GD_p(P, R), IGD_p(P, R)\}, \tag{22}$$

where $GD_p(P, R) = \left(\frac{1}{|P|}\sum_{i=1}^{|P|} d_i^p\right)^{\frac{1}{p}}$ and $IGD_p(P, R) = \left(\frac{1}{|R|}\sum_{j=1}^{|R|} \hat{d}_j^p\right)^{\frac{1}{p}}$, and where d_i and $\hat{d}_j$ are the Euclidean distance from $\vec{x}^i$ to its closest member $\vec{r} \in R$, and the Euclidean distance from $\vec{r}^j$ to its closest member $\vec{x} \in P$, respectively. Here we have chosen $p = 2$. The ideal indicator value is 0, and a low Δ_p value indicates a good approximation of P.

Feasibility Ratio

The feasibility ratio (I_F) indicator refers to the ratio of the number of feasible solutions found in the final approximation P given by a MOEA. Mathematically, this indicator can bee stated as follows.

$$I_F(P) = \frac{P_f}{|P|}, \tag{23}$$

where P_f denotes the number of feasible solutions in P and $|P|$ represents the cardinality of the population P.

5.2. Solving Equality Constrained MOPs with Mathematical Programming Techniques

In this section, we test the NBI technique and apply it to the selected test problems. First, we compute the extreme points of the CHIM by separately optimizing each objective. We take the center of the box defined by each MOP as the initial solution of the optimization process. Then, we compute the CHIM using the previously obtain solution, here we set a partition of 100 well-distributed points. Finally, we solve the NBI subproblem taking into a count each NBI weight. We compute the extreme points of the MOP and the NBI subproblem via *fmincon* Matlab function. Table 1 shows the Δ_2 value and the computational cost, in terms of function evaluations, for each problem. Feval contains the global number of function evaluations required by NBI technique without the cost of each objective optimization process. Feval f_i column contains the number of function evaluations that were required to optimize each objective. Figure A1 of Appendix B shows NBI results for each test problem. Each figure shows the real Pareto front, the computed CHIM and NBI solution.

From Table 1 we observed that in CZDT1, CZDT2, CZDT4 and Das and Dennis Δ_2 value is close to zero which means that NBI adequately solves these problems, on the other hand, the number of function evaluations needed is very high. In the remaining test functions, neither the performance indicator value nor the number of function evaluations is good. Thus, NBI is not capable of solving these MOPs.

Table 1. Values of Δ_2 and number of function calls used by normal boundary intersection (NBI) for the selected test problems.

Function	Δ_2	**Feval**	**Feval f_1**	**Feval f_2**	**Feval f_3**
CZDT1	0.0010	25,759	314	314	-
CZDT2	0.0008	30,221	314	693	-
CZDT3	0.2094	4587	638	830	-
CZDT4	0.0011	22,543	186	186	-
CZDT6	0.0325	6902	106	240	-
D&D	0.1005	84,531	1430	498	-
Eq1-ZDT1	2.7541	89,838	405	4528	-
Eq2-ZDT1	2.6567	87,539	373	5181	-
Eq1-Quad	0.9019	22,679	357	231	649
Eq2-Quad	3.5263	10,449	147	144	4739

5.3. Solving Equality Constrained MOPs with MOEAs

In this section, the proposed approach was compared against four state-of-the-art MOEAs that incorporate different constraint-handling strategies in their environmental selection procedures. Also we include a comparison with a two-stage algorithm that combines the above mention ϵ-NSGA-II (see Section 4.1) and NBI technique.

- NSGA-II. The popular non-dominated sorting genetic algorithm II [50] was adopted in our comparative study. NSGA-II employs a binary tournament-based on feasibility in the mating selection procedure. In order to determine the next generation, the crowding comparison operator considers the feasibility of solutions. In our study, NSGA-II was performed using the standard parameters given by its authors, i.e., $Pc = 1, Pm = 1/n, \eta_c = 20, \eta_m = 20$.
- GDE3. The third evolution step of generalized differential evolution [51] was also adopted in our experimental analysis. GDE3 introduces the concept of constraint-domination explained before to discriminate solutions. GDE3 was employed using $CR = 1$ and $F = 0.5$.
- cMOEA/D-DE. We also adopted the first version of the multi-objective evolutionary algorithm based on decomposition for constraint multi-objective optimization [52]. cMOEA/D-DE utilizes a penalty function in order to satisfy the constraint of the problem. The penalty function is straightforward added to the scalarizing function employed by MOEA/D-DE [53] to approximate the PF of a constrained MOP. cMOEA/D-DE was employed using $CR = 1, F = 0.5, T = \lfloor 0.1 \times N \rfloor, n_r = \lfloor 0.01 \times N \rfloor, Pm = 1/n, \eta = 20, s_1 = 0.01$, and $s_2 = 20$.
- eMOEA/D-DE. A version of MOEA/D-DE based on the ε-constraint method for constrained optimization [54] is also adopted in our experimental study. eMOEA/D-DE employs the ε-constraint method to satisfy the constraints of the problem by obtaining information about feasible solutions in the neighborhood of MOEA/D-DE. Thus, the neighboring solutions are used to defined the ε-constraint value which is dynamically adapted during the search process

of eMOEA/D-DE. eMOEA/D-DE was performed using the standard parameters suggested by its authors, i.e., $CR = 1, F = 0.5, T = \lfloor 0.1 \times N \rfloor, n_r = \lfloor 0.01 \times N \rfloor, Pm = 1/n, \eta = 20, \tau = 0.3$, and $\delta_e = 0.7$.

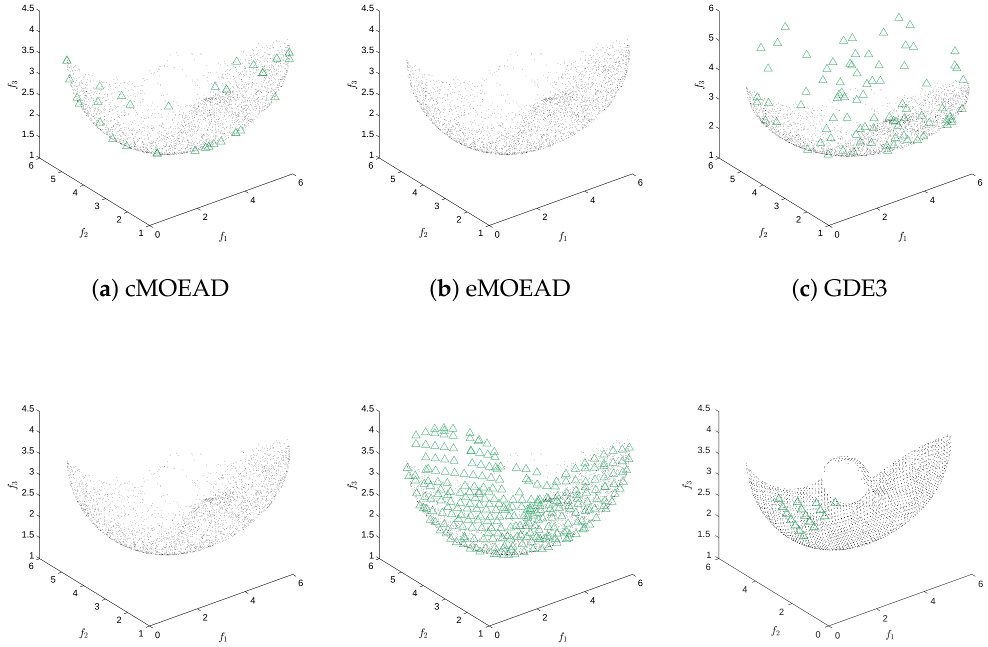
In the above description, N represents the population size which is implicitly defined by the number of subproblems defined by the decomposition-based MOEAs (i.e., cMOEA/D-DE and eMOEA/D-DE). Such subproblems were generated by employing the simplex-lattice design [55] and using the penalty boundary intersection approach (PBI) with a penalty value $\theta = 5$, such as it was suggested by [14]. Therefore, the number of weight vectors is given by $N = C^{M-1}_{H+M-1}$, where M is the number of objective functions. Consequently, the setting of N is controlled by the parameter H. Here, we use $H = 99$ (for two-objective problems) and $H = 23$ (for three-objective problems), i.e., 100 and 300 weight vectors for problems having two and three objectives, respectively.

For each MOP, 30 independent executions were performed with each MOEA.

5.4. Analysis

Our experiments have shown that the new hybrid needs between 15,000 and 17,000 function evaluations (FEs) to obtain good results for the bi-objective problems and 110,000 to 150,000 FEs for the three-objective problems. In order to make the comparison fair, we have set a final budget of 20,000 FEs for all the selected MOEAs on the bi-objective problems and 150,000 FEs for the three-objective problems. For ϵ-NSGA-II/PT we have split the budget for the bi-objective problems into 15,000 FEs for ϵ-NSGA-II and the remaining 5000 FEs for PT (and 100,000 plus 50,000 FEs for three-objective problems). For the realization of PE, we have used Automatic Differentiation to compute the gradients which allows us to express the cost of the continuation method in terms of FEs. For all experiments, we have executed 30 independent runs. We performed the Wilcoxon Test as statistical significance proof to validate the results. For this, we consider the value $\alpha = 0.05$. Based on the test results, for the comparison between ϵ-NSGA-II/PT and any of the chosen MOEAs the symbol "↑" means that the obtained results are statistically significant. The symbol "$--$" means that no Δ_2 value could be computed for the algorithm. This was the case if a MOEA detected no feasible solutions for at least 25 runs.

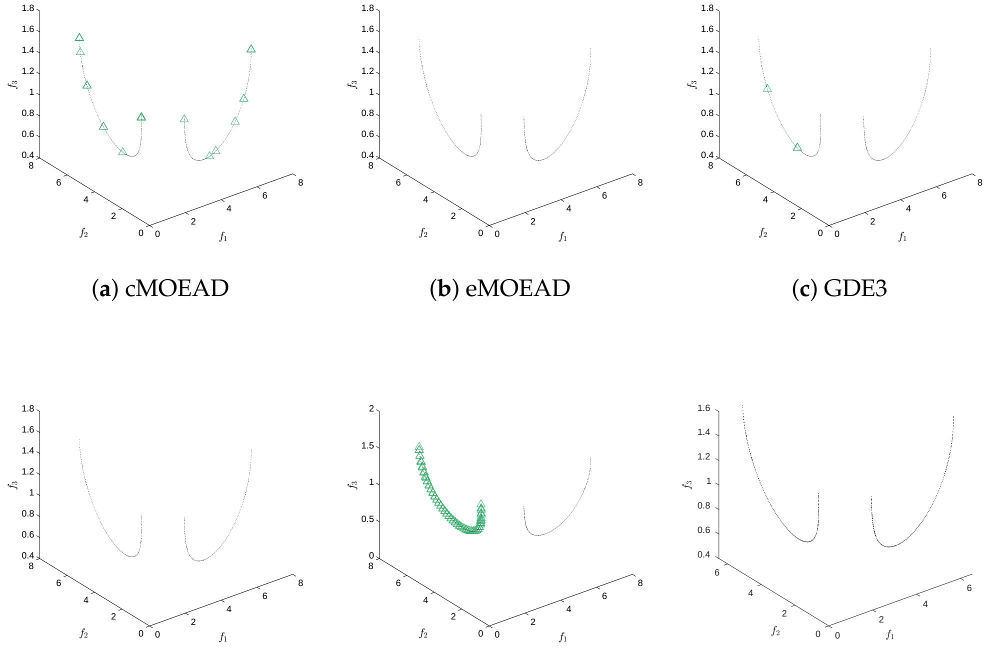
Figures 4 and 5 show the results for Eq1-Quad and Eq2-Quad, respectively. The reader is referred to Appendix C for more figures. The theoretical PF is marked with dots (.) while approximations from the algorithms are marked with triangles (△). The Pareto fronts for the other test problems have been omitted due to space limitations, however, Table 2 shows the indicator values, Δ_2 and I_F, for all test problems. Smaller values of Δ_2 correspond to better qualities of the approximated solution, while larger values for I_F indicate more feasible solutions in the approximation. The best values are displayed in bold for each problem and each indicator. We can see that the new hybrid algorithm significantly outperforms the other algorithms in nine out of the ten test functions considering a compromise between Δ_2 value and the number of the needed function evaluations. Note that in CZDT1, CZDT2, and CZDT6, ϵ-NSGA-II/NBI has better performance that our proposal but it needs more function evaluations in order to achieve these results. In some cases, the MOEAs are not able to find any feasible solution within the given FE budget. ϵ-NSGA-II/PT, however, loses against c-MOEA/D on Eq2-Quad. This is due to the fact that the real Pareto front is disconnected, and in most of the runs, the first stage of our proposal was not able to find adequate solutions near both components. Therefore, in the second stage, we were in most cases only able to compute one of the two components. However, note that all the solutions of the final approximation are feasible in all the independent runs.



(**a**) cMOEAD (**b**) eMOEAD (**c**) GDE3

(**d**) NSGA2 (**e**) ϵ-NSGAII/PT (**f**) ϵ-NSGAII/NBI

Figure 4. Pareto front approximations for Eq1-Quad for the selected MOEAs using a budget of 150, 000 function evaluations.



(**a**) cMOEAD (**b**) eMOEAD (**c**) GDE3

(**d**) NSGA2 (**e**) ϵ-NSGAII/PT (**f**) ϵ-NSGAII/NBI

Figure 5. Pareto front approximations for Eq2-Quad for the selected multi-objective evolutionary algorithms (MOEAs) using a budget of 150, 000 function evaluations.

Table 2. Values of Δ_2 and I_F on the selected test problems.

	Method	**Δ_2**	**I_F**	**Feval**		**Method**	**Δ_p**	**F. Ratio**	
CZDT1	ϵ-NSGA-II/PT	0.0038	**1.0000**	**15,981**	**D&D**	ϵ-NSGA-II/PT	**0.3442**	**1.0000**	**17,175.5**
	(std.dev)	(0.0002)				(std.dev)	(0.6553)		
	ϵ-NSGA-II/NBI↑	**0.0015**	0.9881	26,317		ϵ-NSGA-II/NBI↑	1.2701	0.9887	66,414
	(std.dev)	(0.0014)				(std.dev)	(1.8398)		
	c-MOEA/D	–	0.0000	20,000		c-MOEA/D↑	4.5168	0.0270	20,000
	(std.dev)	(–)				(std.dev)	(2.1485)		
	e-MOEA/D	–	0.0000	20,000		e-MOEA/D	–	0.0000	20,000
	(std.dev)	(–)				(std.dev)	(–)		
	GDE3	–	0.0000	20,000		GDE3	–	0.0000	20,000
	(std.dev)	(–)				(std.dev)	(–)		
	NSGA-II	–	0.0000	20,000		NSGA-II	–	0.0000	20,000
	(std.dev)	(–)				(std.dev)	(–)		
CZDT2	ϵ-NSGA-II/PT	0.0038	**1.0000**	**15,700**	**Eq1-ZDT1**	ϵ-NSGA-II/PT	**0.0158**	**1.0000**	**15,763.2**
	(std.dev)	(0.0002)				(std.dev)	(0.0015)		
	ϵ-NSGA-II/NBI↑	**0.0008**	0.9980	29,298		ϵ-NSGA-II/NBI↑	0.7960	1.0000	161,525
	(std.dev)	(0.0001)				(std.dev)	(0.4744)		
	c-MOEA/D	–	0.0000	20,000		c-MOEA/D ↑	0.4088	0.5060	20,000
	(std.dev)	(–)				(std.dev)	(0.2504)		
	e-MOEA/D	–	0.0000	20,000		e-MOEA/D ↑	0.1683	0.3787	20,000
	(std.dev)	(–)				(std.dev)	(0.0488)		
	GDE3	–	0.0000	20,000		GDE3 ↑	3.0997	0.6653	20,000
	(std.dev)	(–)				(std.dev)	(0.5521)		
	NSGA-II	–	0.0000	20,000		NSGA-II ↑	–	0.0013	20,000
	(std.dev)	(–)				(std.dev)	(–)		
CZDT3	ϵ-NSGA-II/PT	**0.0156**	**1.0000**	**16,235.1**	**Eq2-ZDT1**	ϵ-NSGA-II/PT	**0.1251**	**1.0000**	**16,285.3**
	(std.dev)	(0.0164)				(std.dev)	(0.0428)		
	ϵ-NSGA-II/NBI↑	0.2681	0.8386	201,139		ϵ-NSGA-II/NBI↑	0.6204	1.0000	159,702
	(std.dev)	(0.2298)				(std.dev)	(0.4138)		
	c-MOEA/D	–	0.0000	20,000		c-MOEA/D ↑	0.6624	0.4700	20,000
	(std.dev)	(–)				(std.dev)	(0.2215)		
	e-MOEA/D	–	0.0000	20,000		e-MOEA/D ↑	0.7800	0.4617	20,000
	(std.dev)	(–)				(std.dev)	(0.1235)		
	GDE3	–	0.0000	20,000		GDE3 ↑	3.6144	0.8873	20,000
	(std.dev)	(–)				(std.dev)	(0.5234)		
	NSGA-II	–	0.0000	20,000		NSGA-II ↑	2.4662	0.0037	20,000
	(std.dev)	(–)				(std.dev)	(1.6368)		
CZDT4	ϵ-NSGA-II/PT	**0.0031**	**1.0000**	**16,265.4**	**Eq1-Quad**	ϵ-NSGA-II/PT	**0.1261**	**1.0000**	**149,826.3**
	(std.dev)	(0.0016)				(std.dev)	(0.0043)		
	ϵ-NSGA-II/NBI↑	0.0073	0.9990	529,146		ϵ-NSGA-II/NBI↑	0.1880	0.2478	40,633
	(std.dev)	(0.0055)				(std.dev)	(0.0439)		
	c-MOEA/D	–	0.0000	20,000		c-MOEA/D ↑	0.5714	0.2533	150,000
	(std.dev)	(–)				(std.dev)	(0.0953)		
	e-MOEA/D	–	0.0000	20,000		e-MOEA/D ↑	3.1760	0.0014	150,000
	(std.dev)	(–)				(std.dev)	(0.6012)		
	GDE3	–	0.0000	20,000		GDE3 ↑	0.9133	0.2666	150,000
	(std.dev)	(–)				(std.dev)	(0.0931)		
	NSGA-II	–	0.0000	20,000		NSGA-II	–	0.0001	150,000
	(std.dev)	(–)				(std.dev)	(–)		
CZDT6	ϵ-NSGA-II/PT	0.0884	**1.0000**	**15,739.5**	**Eq2-Quad**	ϵ-NSGA-II/PT	1.9969	**1.0000**	**149,049.2**
	(std.dev)	(0.0180)				(std.dev)	(1.0378)		
	ϵ-NSGA-II/NBI↑	**0.0177**	0.8545	22,362		ϵ-NSGA-II/NBI	–	0.0000	63,299
	(std.dev)	(0.0102)				(std.dev)	()		
	c-MOEA/D	–	0.0000	20,000		c-MOEA/D ↑	**0.4737**	0.1583	150,000
	(std.dev)	(–)				(std.dev)	(0.2000)		
	e-MOEA/D	–	0.0000	20,000		e-MOEA/D	–	0.0000	150,000
	(std.dev)	(–)				(std.dev)	(–)		
	GDE3	–	0.0000	20,000		GDE3 ↑	2.8142	0.0047	150,000
	(std.dev)	(–)				(std.dev)	(1.1008)		
	NSGA-II	–	0.0000	20,000		NSGA-II	–	0.0000	150,000
	(std.dev)	(–)				(std.dev)	(–)		

6. Conclusions

Here the have addressed the treatment of equality constrained MOPs. To this end, we have first proposed three such problems that have different characteristics and that can serve as future benchmark problems. Next, we have proposed a two-phase hybrid evolutionary algorithm, ϵ-NSGA-II/PT, which combines a variant of the well-known and widely used MOEA NSGA-II with the recently proposed continuation method, the Pareto tracer (PT). More precisely, the evolutionary algorithm, ϵ-NSGA-II, computes in a first step a small but ideally well-spread set of solutions around the Pareto front of a given MOP. In the next step, PT takes over to refine the given rough approximation. Empirical results on some benchmark functions that included a comparison against the state-of-the-art have demonstrated that this new hybrid evolutionary algorithm is highly competitive, and yields highly satisfying results using only a moderate budget of function evaluations.

For future work, it might be interesting to reduce the requirement of directly computing the gradient information, e.g., via utilizing approximation strategies like the one proposed in [41], and to apply the new hybrid to the treatment of real-world applications. For instance, our proposal could improve the results presented in [56], where the authors showed that some classic MOEAs have an unsatisfactory performance solving the portfolio problem with complex equality constraints. On the other hand, equality constraints also appear in some single-objective optimization problems (for instance, the development of computer-controlled material [57], the predictive scheduling [58], or the production system designing [59]), and more applications of our proposal could emerge from the extension of such problems to the multi-objective case.

Author Contributions: Conceptualization, O.S. and A.L.; evolutionary algorithms, A.P. and S.Z.-M.; local search, O.C. and L.U. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Conacyt Basic Science project No. 285599, SEP Cinvestav project No. 231, and IPN SIP project No. 20196444.

Acknowledgments: The authors acknowledge funding from the Conacyt Basic Science project No. 285599, SEP Cinvestav project No. 231, and IPN SIP project No. 20196444.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Pareto Tracer (PT)

Here we briefly describe the main steps of the continuation method PT for equality constrained MOPs, more details e.g., on how to treat inequalities can be found in [11].

Starting with a solution x_i, a next solution x_{i+1} is computed in the following two steps: first, the vector ν_μ that is tangential to the KKT set at x_i is chosen via solving

$$\begin{pmatrix} W_\alpha & H^T \\ H & 0 \end{pmatrix} \begin{pmatrix} \nu_\mu \\ \xi \end{pmatrix} = \begin{pmatrix} -J^T\mu \\ 0 \end{pmatrix}, \tag{A1}$$

where J denotes the Jacobian of F at x, $\mu \in \mathbb{R}^k$, $\xi \in \mathbb{R}^p$ and

$$W_\alpha := \sum_{i=1}^{k} \alpha_i \nabla^2 f_i(x) \in \mathbb{R}^{n\times n}, \tag{A2}$$

where $\alpha \in \mathbb{R}^k$ is the Lagrange multiplier, and

$$H := \begin{pmatrix} \nabla h_1(x)^T \\ \vdots \\ \nabla h_p(x)^T \end{pmatrix} \in \mathbb{R}^{p\times n}. \tag{A3}$$

If the ranks of W_α and H are maximal, then ν_μ is determined uniquely. In that case, it holds

$$\sum_{i=1}^{k} \mu_i = 0, \tag{A4}$$

as well as

$$\nu_\mu = -W_\alpha^{-1} J^T \mu. \tag{A5}$$

For a given vector $d \in \mathbb{R}^k$, the direction $\nu_d \in \mathbb{R}^k$ with $J\nu_{\mu_d} = d$ can be computed using (A5):

$$J\nu_{\mu_d} = -JW_\alpha^{-1} J^T \mu_d = d. \tag{A6}$$

Since also (A4) has to hold, μ_d it is determined via solving

$$\begin{pmatrix} -JW_\alpha^{-1} J^T \\ e \end{pmatrix} \mu_d = \begin{pmatrix} d \\ 0 \end{pmatrix}. \tag{A7}$$

The direction vector d can now be chosen to steer along the Pareto front. For this, we have to choose d orthogonal to α: let $\alpha = QR = (q_1, \ldots, q_k)R$ be a QR decomposition of α, then define

$$d_i := q_{i+1}, \quad i = 1, \ldots, k-1, \tag{A8}$$

and select the μ_{d_i}'s via solving (A7). Then, the vectors $\nu_{\mu_{d_i}}$ are tangential to P_Q, and the d_i's are the respective directions in objective space.

Using ν_μ, we can now compute the predictor p_i

$$p_i := x_i + t_i \nu_\mu, \tag{A9}$$

with step size t_i as follows:

$$t_i := \frac{\tau}{\|J\nu_\mu\|}. \tag{A10}$$

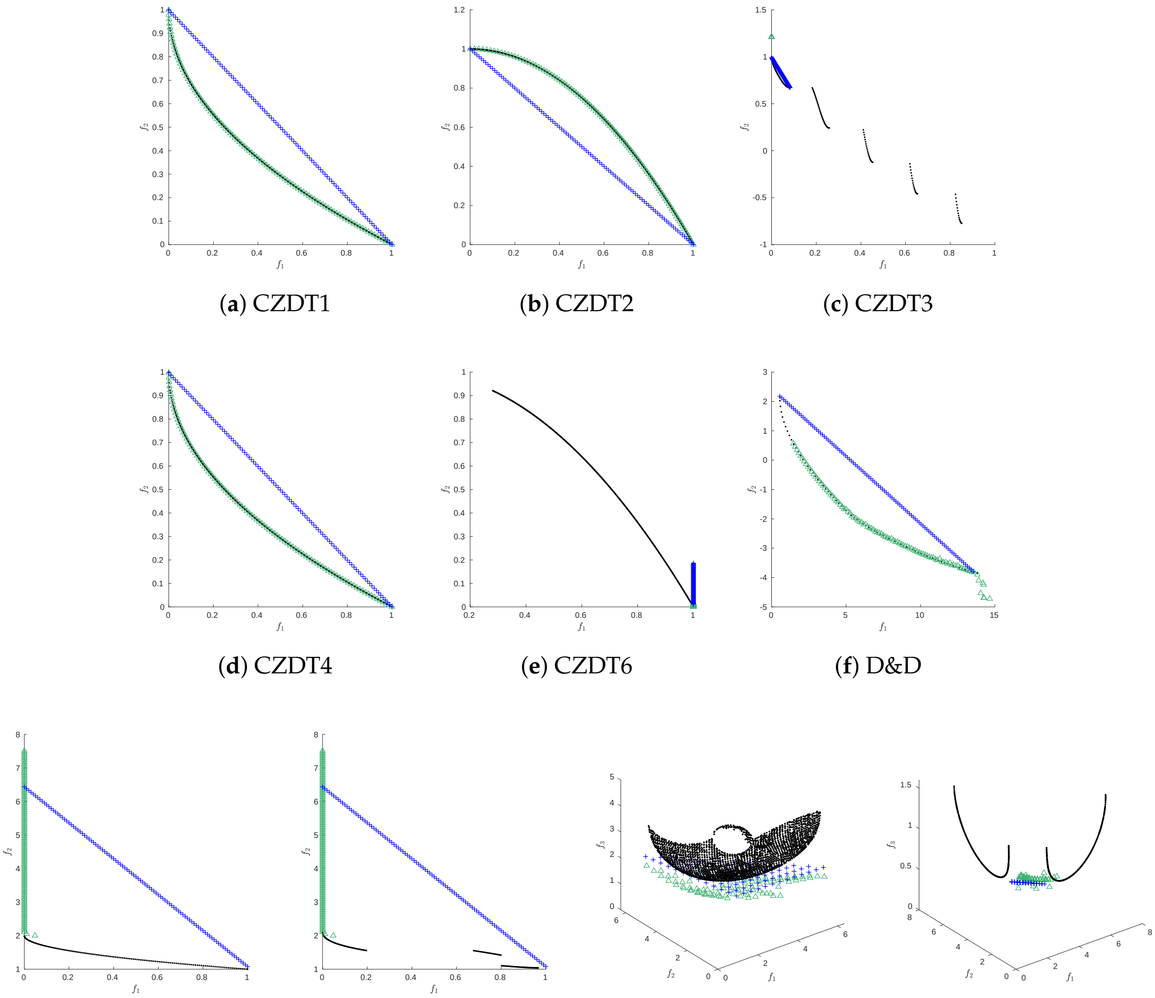
$\tau > 0$ is a user specified value that represents the desired distance between the images of two consecutive solutions in objective space.

In the next step, a Newton method is used to project p_i to the KKT set. The Newton direction is hereby chosen as the solution of

$$\begin{aligned} \min_{(\nu,\delta)\in\mathbb{R}^n\times\mathbb{R}} \quad & \delta \\ \text{s.t.} \quad & \nabla f_i(x)^T \nu + \tfrac{1}{2}\nu^T \nabla^2 f_i(x)\nu \leq \delta, \quad i = 1, \ldots, k, \\ & h_i(x) + \nabla h_i(x)^T \nu = 0, \quad i = 1, \ldots, p. \end{aligned} \tag{A11}$$

Quasi-Newton techniques can be used to realize the algorithm without using Hessians. We will use this implementation here.

Appendix B. Graphical Results for the NBI



(**a**) CZDT1 (**b**) CZDT2 (**c**) CZDT3

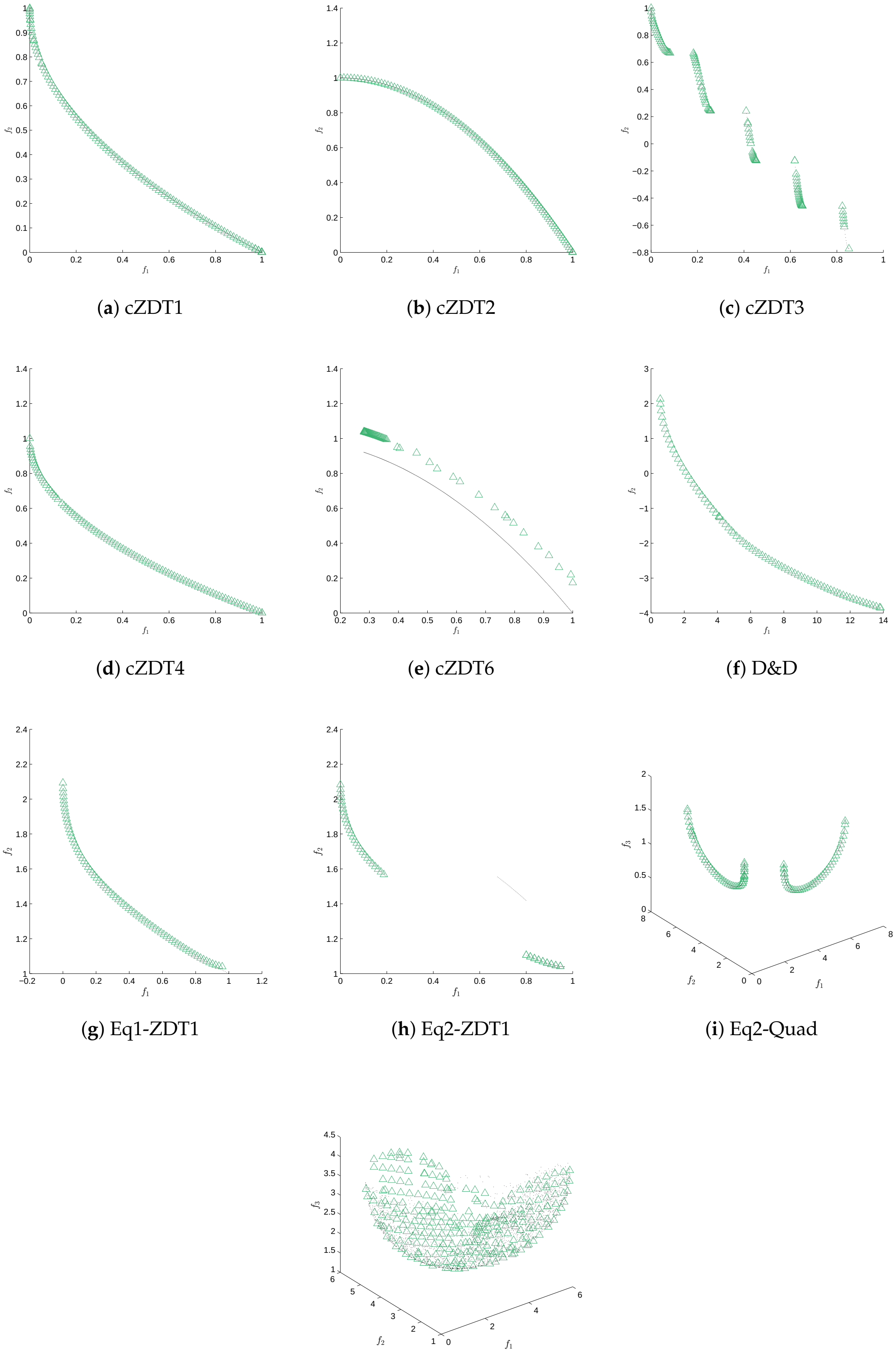
(**d**) CZDT4 (**e**) CZDT6 (**f**) D&D

(**g**) Eq1-ZDT1 (**h**) Eq2-ZDT1 (**i**) Eq1-Quad (**j**) Eq2-Quad

Figure A1. Pareto front approximations for the selected test functions using the NBI method.

Appendix C. Graphical results for the ϵ-NSGAII/PT

Figure A2 shows the numerical results obtained by ϵ-NSGAII/PT on all the ten considered test functions.



(**a**) cZDT1

(**b**) cZDT2

(**c**) cZDT3

(**d**) cZDT4

(**e**) cZDT6

(**f**) D&D

(**g**) Eq1-ZDT1

(**h**) Eq2-ZDT1

(**i**) Eq2-Quad

(**j**) Eq1-Quad

Figure A2. Best result for the ϵ-NSGA-II/PT for all the ten test functions.

Appendix D. Values of γ and η

Table A1 shows the values of γ and η for different dimensions n of the decision variable space. γ and η are used to describe the Pareto sets of Eq1-ZDT1 and Eq2-ZDT1.

Table A1. Values of γ and η.

n	γ	η
16	0.954380	0.863336
17	0.957029	0.848048
18	0.959445	0.832853
19	0.961656	0.817805
20	0.963686	0.802946
21	0.965554	0.788312
22	0.967278	0.773932
23	0.968874	0.759830
24	0.970353	0.746025
25	0.971727	0.732530
26	0.973006	0.719359
27	0.974199	0.706518
28	0.975314	0.694012
29	0.976357	0.681847
30	0.977336	0.670021
31	0.978253	0.658536
32	0.979116	0.647389

References

1. Kalyanmoy, D. *Multi Objective Optimization Using Evolutionary Algorithms*; John Wiley and Sons: Hoboken, NJ, USA, 2001.
2. Coello, C.A.C.; Lamont, G.B.; Van Veldhuizen, D.A. *Evolutionary Algorithms for Solving Multi-Objective Problems*; Springer: Cham, Switzerland, 2007; Volume 5.
3. Ullah, A.S.B.; Sarker, R.; Lokan, C. Handling equality constraints in evolutionary optimization. *Eur. J. Oper. Res.* **2012**, *221*, 480–490. [CrossRef]
4. Ponsich, A.; Jaimes, A.L.; Coello, C.A.C. A Survey on Multiobjective Evolutionary Algorithms for the Solution of the Portfolio Optimization Problem and Other Finance and Economics Applications. *IEEE Trans. Evol. Comput.* **2013**, *17*, 321–344. [CrossRef]
5. Chen, X.L.; Wang, P.H.; Wang, Q.; Dong, Y.H. A Two-Stage strategy to handle equality constraints in ABC-based power economic dispatch problems. *Soft Comput.* **2019**, *23*, 6679–6696. [CrossRef]
6. Liao, Z.; Rittscher, J. A multi-objective supplier selection model under stochastic demand conditions. *Int. J. Prod. Econ.* **2007**, *105*, 150–159. [CrossRef]
7. Bernardino, H.S.; Barbosa, H.J.C.; Angelo, J.S. Differential Evolution with Adaptive Penalty and Tournament Selection for Optimization Including Linear Equality Constraints. In Proceedings of the 2018 IEEE Congress on Evolutionary Computation (CEC), Rio de Janeiro, Brazil, 8–13 July 2018; pp. 1–8. [CrossRef]
8. Barbosa, H.J.C.; Bernardino, H.S.; Angelo, J.S. An improved differential evolution algorithm for optimization including linear equality constraints. *Memetic Comput.* **2018**, *11*, 317–329. [CrossRef]
9. Ghannadpour, S.F.; Noori, S.; Tavakkoli-Moghaddam, R.; Ghoseiri, K. A multi-objective dynamic vehicle routing problem with fuzzy time windows: Model, solution and application. *Appl. Soft Comput.* **2014**, *14*, 504–527. [CrossRef]
10. Cuate, O.; Uribe, L.; Lara, A.; Schütze, O. A benchmark for equality constrained multi-objective optimization. *Swarm Evol. Comput.* **2020**, *52*, 100619. [CrossRef]

11. Martín, A.; Schütze, O. Pareto Tracer: A predictor–corrector method for multi-objective optimization problems. *Eng. Optim.* **2018**, *50*, 516–536. [CrossRef]
12. Cuate, O.; Uribe, L.; Ponsich, A.; Lara, A.; Beltran, F.; Sánchez, A.R.; Schütze, O. A New Hybrid Metaheuristic for Equality Constrained Bi-objective Optimization Problems. In *Evolutionary Multi-Criterion Optimization*; Deb, K., Goodman, E., Coello Coello, C.A., Klamroth, K., Miettinen, K., Mostaghim, S., Reed, P., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 53–65.
13. Jan, M.A.; Khanum, R.A. A study of two penalty-parameterless constraint handling techniques in the framework of MOEA/D. *Appl. Soft Comput.* **2013**, *13*, 128–148. [CrossRef]
14. Zhang, Q.; Li, H. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Trans. Evol. Comput.* **2007**, *11*, 712–731. [CrossRef]
15. Runarsson, T.P.; Yao, X. Stochastic ranking for constrained evolutionary optimization. *IEEE Trans. Evol. Comput.* **2000**, *4*, 284–294. [CrossRef]
16. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [CrossRef]
17. Deb, K.; Pratap, A.; Meyarivan, T. Constrained Test Problems for Multi-objective Evolutionary Optimization. In *Evolutionary Multi-Criterion Optimization*; Zitzler, E., Thiele, L., Deb, K., Coello Coello, C.A., Corne, D., Eds.; Springer: Berlin/Heidelberg, Germany, 2001; pp. 284–298.
18. Zhang, Q.; Zhou, A.; Zhao, S.; Suganthan, P.N.; Liu, W.; Tiwari, S. Multiobjective optimization test instances for the CEC 2009 special session and competition. In *Special Session on Performance Assessment of Multi-Objective Optimization Algorithms*; Technical Report; University of Essex: Colchester, UK; Nanyang Technological University: Singapore, 2008; Volume 264.
19. Ray, T.; Singh, H.K.; Isaacs, A.; Smith, W. Infeasibility Driven Evolutionary Algorithm for Constrained Optimization. In *Constraint-Handling in Evolutionary Optimization*; Mezura-Montes, E., Ed.; Springer: Berlin/Heidelberg, Germany, 2009; pp. 145–165. [CrossRef]
20. Liu, M.; Zou, X.; Chen, Y.; Wu, Z. Performance assessment of DMOEA-DD with CEC 2009 MOEA competition test instances. In Proceedings of the 2009 IEEE Congress on Evolutionary Computation, Trondheim, Norway, 18–21 May 2009; pp. 2913–2918. [CrossRef]
21. Isaacs, A.; Ray, T.; Smith, W. Blessings of maintaining infeasible solutions for constrained multi-objective optimization problems. In Proceedings of the 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence), Hong Kong, China, 1–6 June 2008; pp. 2780–2787. [CrossRef]
22. Asafuddoula, M.; Ray, T.; Sarker, R.; Alam, K. An adaptive constraint handling approach embedded MOEA/D. In Proceedings of the 2012 IEEE Congress on Evolutionary Computation, Brisbane, QLD, Australia, 10–15 June 2012; pp. 1–8. [CrossRef]
23. Fan, Z.; Li, H.; Wei, C.; Li, W.; Huang, H.; Cai, X.; Cai, Z. An improved epsilon constraint handling method embedded in MOEA/D for constrained multi-objective optimization problems. In Proceedings of the 2016 IEEE Symposium Series on Computational Intelligence (SSCI), Athens, Greece, 6–9 December 2016; pp. 1–8. [CrossRef]
24. Takahama, T.; Sakai, S. Constrained Optimization by the ϵ Constrained Differential Evolution with Gradient-Based Mutation and Feasible Elites. In Proceedings of the 2006 IEEE International Conference on Evolutionary Computation, Vancouver, BC, Canada, 16–21 July 2006; pp. 1–8. [CrossRef]
25. Fan, Z.; Li, W.; Cai, X.; Huang, H.; Fang, Y.; You, Y.; Mo, J.; Wei, C.; Goodman, E. An improved epsilon constraint-handling method in MOEA/D for CMOPs with large infeasible regions. *Soft Comput.* **2019**, *23*, 12491–12510. [CrossRef]
26. Fan, Z.; Fang, Y.; Li, W.; Lu, J.; Cai, X.; Wei, C. A comparative study of constrained multi-objective evolutionary algorithms on constrained multi-objective optimization problems. In Proceedings of the 2017 IEEE Congress on Evolutionary Computation (CEC), San Sebastian, Spain, 5–8 June 2017; pp. 209–216. [CrossRef]
27. Ishibuchi, H.; Fukase, T.; Masuyama, N.; Nojima, Y. Dual-grid Model of MOEA/D for Evolutionary Constrained Multiobjective Optimization. In Proceedings of the Genetic and Evolutionary Computation Conference, Kyoto, Japan, 15–19 July 2018; ACM: New York, NY, USA, 2018; pp. 665–672. [CrossRef]
28. Jain, H.; Deb, K. An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point Based Nondominated Sorting Approach, Part II: Handling Constraints and Extending to an Adaptive Approach. *IEEE Trans. Evol. Comput.* **2014**, *18*, 602–622. [CrossRef]

29. Ono, S.; Nakayama, S. Multi-Objective Particle Swarm Optimization for robust optimization and its hybridization with gradient search. In Proceedings of the 2009 IEEE Congress on Evolutionary Computation, Trondheim, Norway, 18–21 May 2009; pp. 1629–1636. [CrossRef]
30. Shim, V.A.; Tan, K.C.; Tan, K.K. A hybrid adaptive evolutionary algorithm in the domination-based and decomposition-based frameworks of multi-objective optimization. In Proceedings of the 2012 IEEE Congress on Evolutionary Computation, Brisbane, QLD, Australia, 10–15 June 2012; pp. 1–8. [CrossRef]
31. Sindhya, K.; Miettinen, K.; Deb, K. A Hybrid Framework for Evolutionary Multi-Objective Optimization. *IEEE Trans. Evol. Comput.* **2013**, *17*, 495–511. [CrossRef]
32. Martínez, S.Z.; Coello Coello, C.A. A direct local search mechanism for decomposition-based multi-objective evolutionary algorithms. In Proceedings of the 2012 IEEE Congress on Evolutionary Computation, Brisbane, QLD, Australia, 10–15 June 2012; pp. 1–8. [CrossRef]
33. Hu, X.; Huang, Z.; Wang, Z. Hybridization of the multi-objective evolutionary algorithms and the gradient-based algorithms. In Proceedings of the 2003 Congress on Evolutionary Computation, Canberra, ACT, Australia, 8–12 December 2003; Volume 2, pp. 870–877. [CrossRef]
34. Hernandez-Diaz, A.G.; Coello Coello, C.A.; Perez, F.; Caballero, R.; Molina, J.; Santana-Quintero, L.V. Seeding the initial population of a multi-objective evolutionary algorithm using gradient-based information. In Proceedings of the 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence), Hong Kong, China, 1–6 June 2008; pp. 1617–1624. [CrossRef]
35. López, A.L.; Coello, C.A.C.; Schütze, O. A painless gradient-assisted multi-objective memetic mechanism for solving continuous bi-objective optimization problems. In Proceedings of the IEEE Congress on Evolutionary Computation, Barcelona, Spain, 18–23 July 2010; pp. 1–8. [CrossRef]
36. Lara, A.; Coello Coello, C.A.; Schutze, O. Using gradient-based information to deal with scalability in multi-objective evolutionary algorithms. In Proceedings of the 2009 IEEE Congress on Evolutionary Computation, Trondheim, Norway, 18–21 May 2009; pp. 16–23. [CrossRef]
37. Martínez-Salazar, I.A.; Molina, J.; Ángel Bello, F.; Gómez, T.; Caballero, R. Solving a bi-objective Transportation Location Routing Problem by metaheuristic algorithms. *Eur. J. Oper. Res.* **2014**, *234*, 25–36. [CrossRef]
38. Braekers, K.; Caris, A.; Janssens, G.K. A Deterministic Annealing Algorithm for a Bi-Objective Full Truckload Vehicle Routing Problem in Drayage Operations. *Procedia- Soc. Behav. Sci.* **2011**, *20*, 344–353. [CrossRef]
39. Kumar, V.S.; Thansekhar, M.; Saravanan, R.; Amali, S.M.J. Solving Multi-objective Vehicle Routing Problem with Time Windows by FAGA. *Procedia Eng.* **2014**, *97*, 2176–2185. [CrossRef]
40. Krink, T.; Paterlini, S. Multiobjective optimization using differential evolution for real-world portfolio optimization. *Comput. Manag. Sci.* **2011**, *8*, 157–179. [CrossRef]
41. Schütze, O.; Alvarado, S.; Segura, C.; Landa, R. Gradient subspace approximation: A direct search method for memetic computing. *Soft Comput.* **2017**, *21*, 6331–6350. [CrossRef]
42. Saha, A.; Ray, T. Equality Constrained Multi-objective optimization. In Proceedings of the 2012 IEEE Congress on Evolutionary Computation, CEC 2012, Brisbane, QLD, Australia, 10–15 June 2012; pp. 1–7.
43. Zitzler, E.; Deb, K.; Thiele, L. Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. *Evol. Comput.* **2000**, *8*, 173–195. [CrossRef] [PubMed]
44. Coello, C.A.C. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: A survey of the state of the art. *Comput. Methods Appl. Mech. Eng.* **2002**, *191*, 1245–1287. [CrossRef]
45. Dellnitz, M.; Schütze, O.; Hestermeyer, T. Covering Pareto Sets by Multilevel Subdivision Techniques. *J. Optim. Theory Appl.* **2005**, *124*, 113–155. [CrossRef]
46. Schütze, O.; Esquivel, X.; Lara, A.; Coello, C.A.C. Using the averaged Hausdorff distance as a performance measure in evolutionary multiobjective optimization. *IEEE Trans. Evol. Comput.* **2012**, *16*, 504–522. [CrossRef]
47. Bogoya, J.M.; Vargas, A.; Cuate, O.; Schütze, O. A (p,q)-Averaged Hausdorff Distance for Arbitrary Measurable Sets. *Math. Comput. Appl.* **2018**, *23*. [CrossRef]
48. Veldhuizen, D.A.V. Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations. Ph.D. Thesis, Department of Electrical and Computer Engineering, Graduate School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, USA, 1999.
49. Coello Coello, C.A.; Cruz Cortés, N. Solving Multiobjective Optimization Problems using an Artificial Immune System. *Genet. Program. Evolvable Mach.* **2005**, *6*, 163–190. [CrossRef]

50. Deb, K.; Thiele, L.; Laumanns, M.; Zitzler, E. Scalable Multi-Objective Optimization Test Problems. In Proceedings of the Congress on Evolutionary Computation (CEC'2002), Honolulu, HI, USA, USA, 12–17 May 2002; IEEE Service Center: Piscataway, NJ, USA, 2002; Volume 1, pp. 825–830.
51. Kukkonen, S.; Lampinen, J. GDE3: The third evolution step of generalized differential evolution. In Proceedings of the IEEE 2005 Congress on Evolutionary Computation (CEC'2005), Edinburgh, UK, 2–5 September 2005; Volume 1, pp. 443–450.
52. Jan, M.A.; Zhang, Q. MOEA/D for constrained multiobjective optimization: Some preliminary experimental results. In Proceedings of the 2010 UK Workshop on Computational Intelligence (UKCI), Colchester, UK, 8–10 September 2010; pp. 1–6.
53. Li, H.; Zhang, Q. Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II. *IEEE Trans. Evol. Comput.* **2009**, *13*, 284–302. [CrossRef]
54. Zapotecas Martínez, S.; Coello Coello, C.A. A Multi-objective Evolutionary Algorithm based on Decomposition for Constrained Multi-objective Optimization. In Proceedings of the 2014 IEEE Congress on Evolutionary Computation (CEC'2014), Beijing, China, 6–11 July 2014; pp. 429–436. [CrossRef]
55. Scheffé, H. Experiments with Mixtures. *J. R. Stat. Soc.* **1958**, *20*, 344–360. [CrossRef]
56. Skolpadungket, P.; Dahal, K.; Harnpornchai, N. Portfolio optimization using multi-obj ective genetic algorithms. In Proceedings of the 2007 IEEE Congress on Evolutionary Computation, Singapore, 25–28 September 2007; pp. 516–523. [CrossRef]
57. Gola, A.; Kłosowski, G. Development of computer-controlled material handling model by means of fuzzy logic and genetic algorithms. *Neurocomputing* **2019**, *338*, 381–392. [CrossRef]
58. Sobaszek, Ł.; Gola, A.; Świć, A. Predictive Scheduling as a Part of Intelligent Job Scheduling System. In *Intelligent Systems in Production Engineering and Maintenance–ISPEM 2017*; Burduk, A., Mazurkiewicz, D., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 358–367.
59. Plinta, D.; Krajčovič, M. Production System Designing with the Use of Digital Factory and Augmented Reality Technologies. In *Progress in Automation, Robotics and Measuring Techniques*; Szewczyk, R., Zieliński, C., Kaliczyńska, M., Eds.; Springer International Publishing: Cham, Switzerland, 2015; pp. 187–196.