Using ϵ-Dominance for Hidden and Degenerated Pareto-Fronts

Heiner Zille
Institute of Knowledge and Language Engineering
University of Magdeburg, Germany
Email: heiner.zille@ovgu.de

Sanaz Mostaghim
Institute of Knowledge and Language Engineering
University of Magdeburg, Germany
Email: sanaz.mostaghim@ovgu.de

***Abstract*—Scalable multi-objective test problems are known to be useful in testing and analyzing the abilities of algorithms. In this paper we focus on test problems with degenerated Pareto-fronts and provide an in-depth insight into the properties of some problems which show these characteristics. In some of the problems with degenerated fronts such as Distance Minimization Problem (DMP) with the Manhattan metric, it is very difficult to dominate some of the non-optimal solutions as the optimal solutions are hidden within a set of so called pseudo-optimal solutions. Hence the algorithms based on Pareto-domination criterion are shown to be inefficient. In this paper, we explore the pseudo-optimal solutions and examine how and why the use of ϵ-dominance can help to achieve a better approximation of the hidden Pareto-fronts or of degenerated fronts in general. We compare the performance of the ϵ-MOEA with 3 other algorithms (NSGA-II, NSGA-III and MOEA/D) and show that ϵ-dominance performs better when dealing with pseudo-optimal kind of solutions. Furthermore, we analyze the performance on the WFG3 test problem and illustrate the advantages and disadvantages of ϵ-dominance for this degenerated problem.**

I. Introduction

Real-world multi-objective problems usually contain Pareto-fronts of various shapes and geometries such as convex, concave, disconnected and degenerated. In this sense, the existing test problems resemble these properties and provide a platform to test the ability of the algorithms [1]. The major goal of many of the existing multi-objective evolutionary algorithms is to find a set of diverse solutions along the Pareto-front. In the literature, many existing approaches have been tested and analyzed on various test problems. Nevertheless, to our knowledge, the problems with degenerated Pareto-fronts have not been analyzed as much as the other test problems. In a previous work, we found that the Distance Minimization Problem (DMP) using the Manhattan metric has a degenerated Pareto-front [2] and a certain special structure of domination. Therefore, the goal of this paper is to provide an analysis of the properties of optimal solutions in certain problems with degenerated Pareto-fronts. The difficulty in finding optimal solutions for such problems has also been stated in the literature e.g., by Huband et al. [1] and Deb et. al. [3]. Examples for test Problems with degenerated fronts are the mentioned Distance Minimization Problem (DMP) using the manhattan metric, the WFG3 problem [1] and the 3-objective instance of the DTLZ5 problem [3].

In [2], we analyzed the properties of the DMP with Manhattan metric and found out that in such problems when approaching the Pareto-optimal area in the decision space, it gets very difficult to find dominating solutions to the current population since the amount of dominating solutions in the neighborhood of the current population gets smaller. In such a case, if there is only a sparse set of solutions which dominates the current solution sets members, the probability of finding members of this dominating set is relatively small, particularly by using genetic operators. Additionally, even when Pareto-optimal solutions are already found within the current population, these Pareto-optimal solutions might still be surrounded by a large set of solutions that are non-dominated to them, even though they are not optimal. In this work, we aim to examine this kind of solutions in further detail.

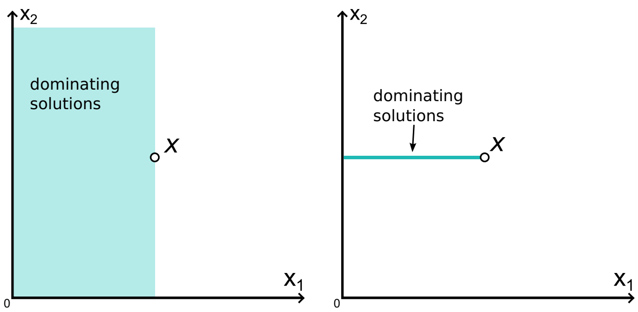
In this paper, we introduce *pseudo-optimal* solutions and provide a concept for *hidden Pareto-fronts* which are optimal solutions surrounded by sets of indifferent *pseudo-optimal* solutions. Additionally, we will deal with the concept of such hard to dominate solutions and the corresponding Pareto-optimal fronts. A special focus is to the performance of the ϵ-MOEA, which uses ϵ-dominance and should in theory be suitable to overcome the weaknesses of Pareto-dominance that were described in [2]. For the analysis, we take the DMP with the Manhattan metric and introduce a simple toy problem for which we can specify the properties for the degenerated Pareto-front. In addition, we show some results on the WFG3 problem [1], to investigate whether the found results on problems with *pseudo-optimal* solutions also reproduce for other problems with degenerated Pareto-fronts. We analyze the performance of the algorithms NSGA-II [4], NSGA-III [5], MOEA/D [6] and ϵ-MOEA [7] on these problems with the goal to identify the methods which help the algorithms to deal with this kind of domination structure.

The remainder of this paper is structured as follows. First we introduce the concept of the *pseudo-optimal* solutions and *hidden Pareto-fronts*, by describing the properties of hard-to-dominate solutions in detail. In Section III, a description of the Manhattan metric based DMP and a toy problem are given with the goal to understand the structure of the domination in both of these problems. Section IV contains the experiments and the comparisons and the paper is concluded in section V.

II. Pseudo-optimal Solutions

The goal of this section is to provide some fundamental definitions for the solutions of a multi-objective problem. In the literature there are several definitions such as domination, strong domination or weak domination [8]. Here we want

978-1-4799-7560-0/15 $31.00 © 2015 IEEE
DOI 10.1109/SSCI.2015.125



(a) Non-pseudo-optimal solution $\vec{x}$ and the area of dominating solutions in P_1

(b) Pseudo-optimal solution $\vec{x}$ and the area of dominating solutions in P_2

Fig. 1. Non-pseudo-optimal and pseudo-optimal solutions in P_1 and P_2 and the areas that dominate them (example).

to further differentiate between the solutions in terms of the existing structures in the decision space, which have strong impact on the strength of the domination.

Pseudo-optimal Solution: We call a solution $\vec{x} \in \mathbb{R}^n$ to be *pseudo-optimal*, if $\vec{x}$ is not Pareto-optimal and it is hard to find a solution which can dominate $\vec{x}$ in the neighborhood of $\vec{x}$.

This means that a solution can be pseudo-optimal when there are dominating solutions in its neighborhood, but the probability of randomly picking them from the neighborhood is very low. Of course the definition of *low* in this case is quite variable. It is very unlikely to find the dominating solutions in a neighborhood, if we have a small number of them. For instance, let us consider a solution $\vec{x} = (x_1, x_2) \in \mathbb{R}^2$ and two different optimization problems P_1 and P_2 with the (weak) Pareto-dominance relation $\preceq_p$. Without knowing the exact fitness function, let us assume that the domination structures of both problems look like following:

$$\begin{aligned} P_1: &\quad \vec{y} \preceq_p \vec{x} \iff y_1 \leq x_1 \\ P_2: &\quad \vec{y} \preceq_p \vec{x} \iff y_1 \leq x_1 \wedge y_2 = x_2 \end{aligned} \tag{1}$$

In P_1, $\vec{x}$ is dominated by any other solution that has a smaller value for the first decision variable. In P_2, this is only the case if also the value of the second decision variable is the same. Now we can compute the probabilities of finding a dominating solution in both problems, using the neighborhood $H(\vec{x}) := \{\vec{z} \in \mathbb{R}^2 | z_1 = x_1 + \Delta_1 \wedge z_2 = x_2 + \Delta_2, \Delta_1, \Delta_2 \in [-\epsilon, \epsilon]\}$:

$$\begin{aligned} P_1: \quad & Prob(\vec{y} \preceq_p \vec{x}) \\ &= Prob(y_1 \leq x_1) \\ &= Prob(x_1 + \Delta_1 \leq x_1) \\ &= Prob(\Delta_1 \leq 0) \\ &= 0.5 \\ \\ P_2: \quad & Prob(\vec{y} \preceq_p \vec{x}) \\ &= Prob(y_1 \leq x_1 \wedge y_2 = x_2) \\ &= Prob(y_1 \leq x_1) * Prob(y_2 = x_2) \\ &= 0.5 * Prob(x_2 + \Delta_2 = x_2) \\ &= 0.5 * Prob(\Delta_2 = 0) \\ &= 0.5 * 0 \\ &= 0 \end{aligned} \tag{2}$$

Figure 1 illustrates this example. We can observe that it is unlikely to find dominating solutions for $\vec{x}$ in P_2 using a random search in its neighborhood. For that reason, since we are dealing with continuous decision variables and operators in evolutionary algorithms that involve randomness to create new solutions during recombination and mutation, we can assume that the possibility to exactly create a dominating solution in the above example of P_2 is almost the same as for a random picking mechanism to obtain these solutions. In this example, for any randomly selected solution $\vec{p}$ in the neighborhood of $\vec{x}$, $\vec{x}$ is Pareto-dominated by $\vec{p}$ with a probability of 0. So the set of dominating solutions in the neighborhood is actually in this case 1-dimensional, which is found in a 2-dimensional neighborhood with a probability of 0.

The above example indicates that pseudo-optimal solutions are unlikely to be dominated, if they are used to produce new solutions close to them by genetic operators.

The difficulty arises, when the true Pareto-optimal solutions are located within the area of the fitness landscape containing pseudo-optimal solutions. Since pseudo-optimal solutions are not optimal, we can theoretically find dominating solutions around them and eventually approach the true Pareto-optimal front. However, since newly created solutions are unlikely to dominate any existing ones, it is very difficult to distinguish between pseudo-optimal and optimal solutions. In that case, we have a so called *hidden Pareto-front*.

Hidden Pareto-optimal Solution: A Pareto-optimal solution $\vec{x}^*$ is a *hidden Pareto-optimal solution*, if every other solution $\vec{y}$ in the neighborhood of $\vec{x}^*$ is either Pareto-optimal, or pseudo-optimal. If all of the Pareto-optimal solutions of a Problem are hidden, we speak of a *hidden Pareto-optimal front* of the Problem.

One problem that has this kind of pseudo-optimal solutions and an at least partially hidden Pareto-optimal front is the Manhattan-metric based Distance Minimization Problem (Manhattan-DMP). This problem will be discussed in the next section. The actual Pareto-optimal set in the decision space is surrounded by a large set of pseudo-optimal solutions, that are not Pareto-optimal, but can only be dominated by a very specific set of solutions and only 1 (numerically exact) Pareto-optimal solution. Due to that structure, the algorithms used in [2] that were based on Pareto-dominance failed to approximate the Pareto-optimal front. Zille and Mostaghim studied the ability of 3 different algorithms (among them NSGA-II [4] and MOEA/D [6]) to solve 2-variable instances of the Manhattan metric DMP, and found that the algorithms which use Pareto-dominance fail to approximate the true Pareto-set.

III. Problems with degenerated Pareto fronts

In this section we briefly study and introduce problems with degenerated fronts which contain Pseudo-optimal solutions and hidden Pareto-fronts.

A. The Manhattan Distance Minimization Problem

The Distance Minimization Problem (DMP) is a scalable multi-objective optimization problem which contains a set of predefined so-called *objective-points* $\{\vec{O}_1, .., \vec{O}_m\}$ with coordinates $\vec{O}_i = (o_{i1}, .., o_{in})^T$ in a n-dimensional decision space [2], [9], [10], [11], [12]. The number of objective-points corresponds to the number of objectives (m). The goal of the DMP is to find a set of solution vectors ($\in \mathbb{R}^n$) in the decision space which have a minimum distance to all of the objective-points. It is formulated as follows:

$$\begin{array}{lll} min & f(\vec{x}) = (f_1(\vec{x}), f_2(\vec{x}), ..., f_m(\vec{x}))^T & \\ s.t. & f_i = dist(\vec{x}, \vec{O}_i) & \forall i = 1, .., m \\ & x_j \leq x_{max,j} & \forall j = 1, .., n \\ & x_j \geq x_{min,j} & \forall j = 1, .., n \end{array} \quad (3)$$

For measuring the distances between two points in the decision space, we select the $dist$ function to be the Manhattan metric ($p-1$ metric) induced by the $p-1$ norm as follows:

$$dist_1(\vec{a}, \vec{b}) := \| \vec{a} - \vec{b} \|_1 = \sum_{i=1}^{n} |a_i - b_i| \quad (4)$$

The Pareto-optimal front of this problem is partially hidden according to the definition in Section II. Most of the Pareto-optimal solutions are surrounded by pseudo-optimal solutions and therefore Pareto-dominance based algorithms have problems to find them, even in a very small 2-variable problem. For an analysis about the shapes and properties of the Manhattan-based DMP refer to [2].

B. Toy Problem

Since the composition of the Pareto-optimal areas in Manhattan-based DMP is quite complex, we introduce a simple problem that has the desired pseudo-optimal areas and is very easy to scale and visualize. In this case, we can investigate the performance of the algorithms when dealing with the pseudo-optimal solutions and hidden Pareto-optimal fronts. Let us consider the following simple example problem with 3 objective functions and 2 decision variables:

$$\begin{array}{ll} min & f_1(\vec{x}) = |A - x_1| \\ & f_2(\vec{x}) = |B - x_1| \\ & f_3(\vec{x}) = |C - x_2| \\ s.t. & x_1, x_2 \leq 10 \\ & x_1, x_2 \geq 0 \end{array} \quad (5)$$

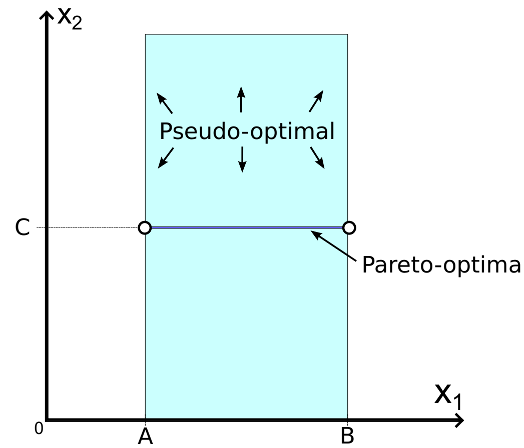


Fig. 2. Decision space of the example problem with Pareto-optimal solutions and hard-to-dominate solutions

where A, B and C are constant values. Figure 2 illustrates the decision space of this problem for 2 decision variables. The Pareto-optimal set of solutions in the decision space is given by $x_2 = C \wedge x_1 \in [A, B]$. This Pareto-front is hidden within pseudo-optimal solutions. Consider the constants to be $A = 4$, $B = 6$ and $C = 5$. Taking two solutions $x = (x_1, x_2) = (4.5, 8)$ and $y = (y_1, y_2) = (4.5001, 6)$, it is clear that solution y is located closer to the true Pareto-front than solution x. However, these solutions do not dominate each other in a Pareto-sense. Hence, the algorithms using the Pareto-domination criterion consider these solutions to be equal in terms of quality.

In this simple test problem, only the objectives f_1 and f_2 are conflicting and therefore the other function (f_3) can be separately optimized. Nevertheless, even though the objectives are not conflicting, the attempt to optimize them at the same time makes the problem much harder to solve. This problem is scalable to any arbitrary number of decision variables (n) and objective functions (m). The generalized problems looks like the following:

$$\begin{array}{ll} min & f_1(\vec{x}) = |A - x_1| \\ & f_2(\vec{x}) = |B - x_1| \\ & f_i(\vec{x}) = |C - x_{i-1}| \quad \forall i = 3..m-1 \\ & f_m(\vec{x}) = \sum_{k=m}^{n} |C - x_k| \\ s.t. & x_j \leq 10 \quad \forall j = 1..n \\ & x_j \geq 0 \quad \forall j = 1..n \end{array} \quad (6)$$

The Pareto-optimal set of solutions in the decision space and the corresponding Pareto-optimal front in the objective space can be described as following:

$$PS: \quad x_1 \in [A, B]$$
$$x_i = C \quad \forall i = 2..n$$

$$PF: \quad f_1 \in [0, |B - A|] \tag{7}$$
$$f_2 \in [0, |B - A|]$$
$$f_1 + f_2 = |B - A|$$
$$f_j = 0 \quad \forall j = 3..m$$

As we could observe in this problem as well as in the Manhattan-based DMP, the hidden Pareto-fronts are both degenerated in a way that the Pareto-optimal set of solutions in the search space (partly) consists of areas of smaller dimension than the decision variable space. Most parts of the Pareto-optimal front of the 2-variable Manhattan-based DMP form lines instead of volumes, which makes it even more difficult for the algorithms to obtain the Pareto-front. The Pareto-optimal front in the objective space is also a set of 3 lines which are 1-dimensional subsets of the 3-dimensional objective space.

IV. EXPERIMENTS

After defining the properties in Section II and the test problems in Section III, we aim to analyze different multi-objective evolutionary algorithms to investigate the methodologies which help deal with problems with pseudo-optimal solutions, hidden Pareto-fronts and degenerated Pareto-fronts. The goal of the experiments is not to generally compare the algorithms with each other. Here we are interested to investigate the mechanisms and therefore the analysis on the simple test problems is used. Furthermore, besides the typical evaluation mechanisms, we aim to visualize the 2 dimensional decision spaces.

Among the existing algorithms, we have selected 4 algorithms with different mechanisms for obtaining the Pareto-optimal solutions:

- **NSGA-II** The NSGA-II is an evolutionary algorithm developed by Deb et al. [4] and uses non-dominated sorting and crowding distance.
- **NSGA-III** The new version of the non-dominated sorting algorithm was developed in 2014 and focuses on reference-direction vectors for diversity instead of crowding, and also omits a selection operator [3].
- **MOEA/D** This evolutionary algorithm completely omits Pareto-dominance and uses an aggregation function with weight vectors to solve multiple single-objective problems instead of one multi-objective problem [6].
- **ϵ-MOEA** The ϵ-MOEA uses two components. ϵ-dominance is used as the domination criteria, while a cell-structure is applied to the solution set to maintain diversity. The Pareto-dominance is used for comparison within the same ϵ-cell [7].

For the analysis, we measure the Generational Distance (GD) and Inverted Generational Distance (IGD) values of the obtained non-dominated fronts. We perform 50 independent runs for each algorithm on the 2-variable Manhattan DMP and different instances of the toy problem, since its Pareto-optimal fronts are easier to determine in higher dimensions. In addition, since we are interested to know if the possible advantages of the mechanisms transfer to degenerated Pareto-fronts in general, we use different instances of the WFG3 problem [1].

The number of objectives is set to be 3 in all tested problems. The number of decision variables is varied. The WFG3 problem splits the decision variables artificially into position-related and distance-related variables. We split these in a 1/3 to 2/3 relation, which means a 30 variable instance has 10 position-related and 20 distance-related variables. Also, the WFG3 function requires the position-related variables to be a multiple of $m - 1$ (m is the number of objectives) and the distance-related variables to be an even number.

In addition to the GD and IGD values we want to show some randomly chosen runs as plots in the decision and objective space of the problems to visualize the distribution of solutions and use this for the analysis.

A. Parameter settings

For all algorithms we use 120,000 function evaluations as the stopping criterion. Our experiments are performed with the MOEA-framework [13], and we use the standard MOEA-framework implementations and parameter settings for the algorithms and operators (SBX crossover, polynomial mutation, crossover-rate 1.0, crossover-distribution-index 15.0, mutation-rate $1/n$, mutation-distribution-index 20.0). The aggregation function for the MOEA/D is the Chebyshev method.

We set the population sizes of the MOEA/D, the NSGA-II and the NSGA-III to 100. The ϵ-values of the ϵ-MOEA are experimentally adjusted beforehand to values that will result in final population sizes close to the population sizes of the other algorithms. Due to the search process of the ϵ-MOEA, the final population sizes may vary a bit, so that some runs yield a population size of 99, some of 104. It is known that different population sizes may result in less reliable numbers for GD and IGD values for comparison. However, since the fluctuations are minimal compared to the population size of 100, it can be assumed that the numbers obtained are sufficient for comparison of the algorithms performance.

In our experiments, we use the instance of the DMP created in the same way as in [2], where the 3 objective-points are distributed evenly around a center and the initial offset is $\alpha_1 = \pi/4$. For the toy problem, $A = 4$, $B = 6$ and $C = 5$.

B. Results and Analysis

The Tables I and II give an overview of the average obtained GD and IGD values as well as the corresponding standard errors for all tested problems. Additionally, the Figures 4 to 10 show the plots of the obtained fronts and the true Pareto-front of some instances of the problems in objective and also - where possible - in decision spaces.

Manhatten based DMP: The results on the Manhattan based DMP in Tables I and II indicate that the ϵ-MOEA has advantages over the two algorithms which use non-dominated sorting (NSGA-II and NSGA-III). The performance of the MOEA/D is similar to the ϵ-MOEA. The ϵ-MOEA obtains on average

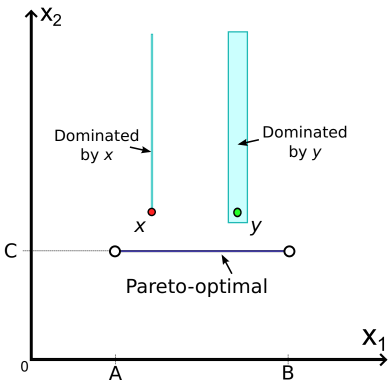
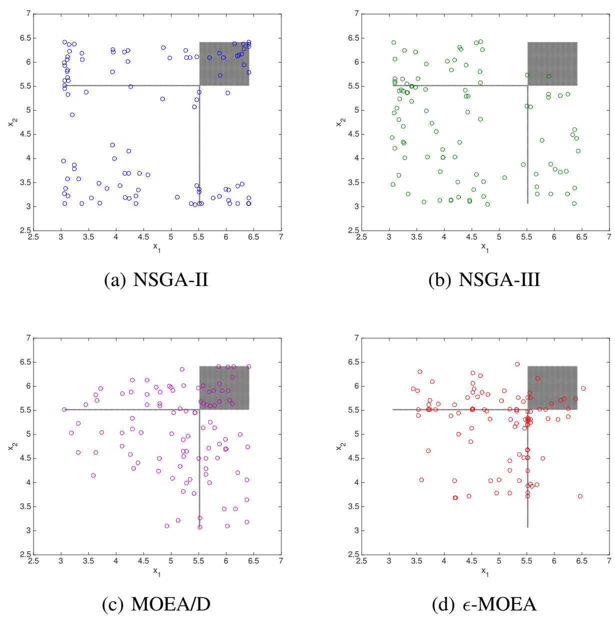


Fig. 3. Areas dominated by solution x without and by solution y with ϵ-domination.



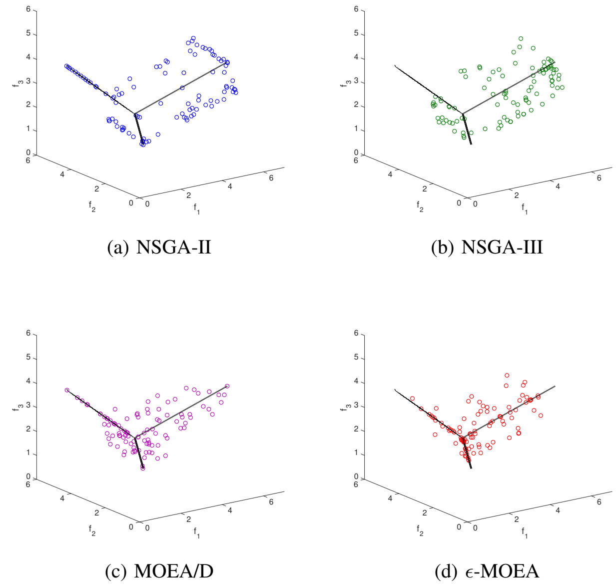
(a) NSGA-II (b) NSGA-III (c) MOEA/D (d) ϵ-MOEA

Fig. 4. Decision space of the 2-variable Manhattan-DMP with 3 objectives. The Pareto-front is given as solid lines and areas.

solutions closer to the Pareto-front (higher GD-value), but the MOEA/D algorithm achieves only slightly better diversity. This can also be seen in the plots of 1 example run of all 4 algorithm in the Figures 4 and 5.

The MOEA/D is able to achieve a better approximation of the Pareto-optimal front than the NSGA-II algorithm, since it does not use Pareto-dominance. This is consistent with the results found in [2]. The NSGA-III performs similar to the NSGA-II, since it also uses the non-dominated sorting approach. However, we can see that the distribution of the solutions is slightly better, which might be an effect of the reference-direction approach used in NSGA-III, that resembles the concept of the weight vectors used in MOEA/D to achieve diversity.

The usage of ϵ-dominance is especially suitable to counter the properties of the pseudo-optimal solutions. Since they can



(a) NSGA-II (b) NSGA-III (c) MOEA/D (d) ϵ-MOEA

Fig. 5. Objective space of the 2-variable Manhattan-DMP with 3 objectives. The Pareto-front is given as solid lines.

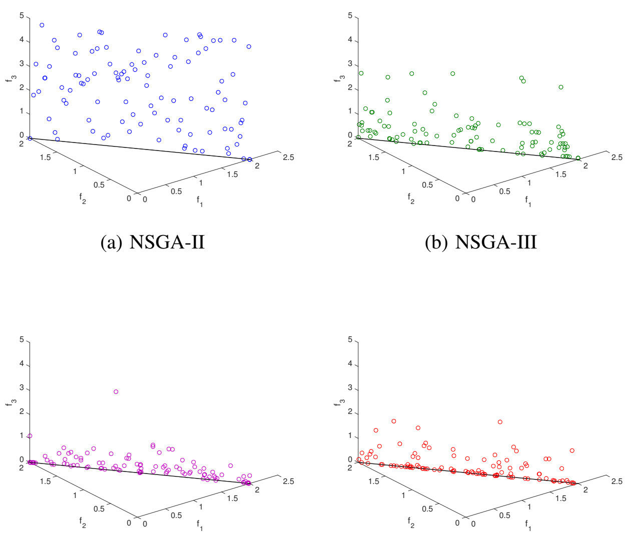
TABLE I. AVERAGE GD VALUES AND STANDARD ERRORS (50 RUNS) FOR DIFFERENT PROBLEMS WITH 3 OBJECTIVES (120,000 EVALUATIONS). BEST RESULTS ARE MARKED IN BOLD.

Problem	n	NSGA-II	NSGA-III	MOEA/D	ϵ-MOEA
DMP	2	0.0336 (± 1,66e-04)	0.0283 (± 4,93e-04)	0.0149 (± 2,94e-04)	**0.0136** (± 2,79e-04)
Toy Problem	2	0.2764 (± 1,33e-03)	0.1705 (± 7,67e-03)	0.0548 (± 1,40e-03)	**0.0531** (± 1,16e-03)
	20	1.3179 (± 2,32e-02)	0.4871 (± 9,66e-03)	0.1369 (± 6,72e-03)	**0.0981** (± 2,69e-03)
	40	1.3891 (± 2,17e-02)	0.6717 (± 1,15e-02)	0.1700 (± 8,18e-03)	**0.1002** (± 2,05e-03)
	100	1.8655 (± 2,95e-02)	1.1355 (± 1,56e-02)	0.3455 (± 2,84e-02)	**0.1017** (± 1,98e-03)
WFG3	2 - 4	0.0811 (± 4,56e-04)	0.0828 (± 2,94e-03)	0.0817 (± 4,57e-04)	**0.0422** (± 7,71e-04)
	10 - 20	0.0719 (± 5,80e-04)	0.0068 (± 1,87e-04)	0.0827 (± 4,74e-04)	**0.0063** (± 4,26e-04)
	20 - 40	0.0454 (± 7,26e-04)	**0.0091** (± 2,61e-04)	0.0844 (± 5,54e-04)	0.0182 (± 7,46e-04)
	40 - 60	0.0229 (± 7,81e-04)	**0.0133** (± 4,80e-04)	0.0886 (± 5,36e-04)	0.0356 (± 1,75e-03)

only be dominated by a specific and small subset of neighboring solutions with classical Pareto-dominance, the usage of ϵ-dominance allows a solution to dominate a range of solutions normally indifferent to it. By enabling a solution to dominate more solutions within its neighborhood than just in a normal Pareto sense, we eliminate the need to numerically exactly find the solutions that dominate the non-optimal ones. In a neighborhood that contains only a very small set of dominating solutions, the use of ϵ-dominance artificially creates a solution vector that might not be achievable by the normal search process, since this solution is able to dominate a volume of solutions that can normally not be dominated by one solution alone. We depicted this situation for the toy problem and two decision variables in Figure 3, and the same structure is also the situation in the Manhattan-DMP. We see that the use ϵ-dominance allows a solution to dominate a range of solutions

TABLE II. Average IGD values and standard errors (50 runs) for different problems with 3 objectives (120,000 evaluations). Best results are marked in bold.

Problem	n	NSGA-II	NSGA-III	MOEA/D	ϵ-MOEA
DMP	2	0.0029 (± 6,98e-05)	0.0024 (± 6,34e-05)	**0.0013** (± 2,55e-05)	0.0014 (± 3,40e-05)
Toy Problem	2	0.0074 (± 2,43e-04)	0.0045 (± 1,65e-04)	0.0021 (± 5,45e-05)	**0.0009** (± 1,07e-05)
	20	0.0217 (± 5,18e-04)	0.0992 (± 2,93e-03)	0.0040 (± 9,12e-05)	**0.0011** (± 1,06e-05)
	40	0.0601 (± 1,10e-03)	0.1714 (± 3,40e-03)	0.0064 (± 1,68e-04)	**0.0011** (± 8,48e-06)
	100	0.3263 (± 4,54e-03)	0.3767 (± 6,08e-03)	0.0349 (± 6,08e-04)	**0.0027** (± 4,29e-05)
WFG3	2 - 4	0.0041 (± 1,16e-04)	0.0042 (± 9,70e-05)	**0.0019** (± 4,78e-05)	0.0106 (± 2,62e-04)
	10 - 20	0.0047 (± 1,10e-04)	0.0048 (± 2,13e-04)	**0.0024** (± 5,16e-05)	0.0328 (± 2,08e-04)
	20 - 40	0.0082 (± 2,12e-04)	0.0156 (± 1,04e-03)	**0.0053** (± 9,72e-05)	0.0635 (± 2,40e-03)
	40 - 60	0.0152 (± 2,72e-04)	0.0354 (± 1,50e-03)	**0.0128** (± 2,05e-04)	0.0759 (± 6,49e-04)



(a) NSGA-II (b) NSGA-III

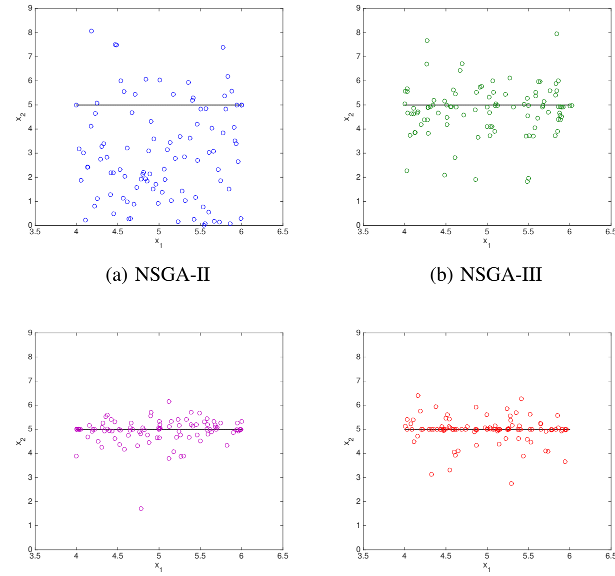
(c) MOEA/D (d) ϵ-MOEA

Fig. 6. Objective space of the 2-variable toy problem with 3 objectives. The Pareto-front is given as a solid line.

which eliminates the difficult structure of the fitness landscape.

Toy Problem: If we take a look at the toy problem, we can see that the same effect as in the DMP is visible for all numbers of decision variables. The algorithm that uses ϵ-dominance can outperform the other algorithms, especially the ones that use Pareto-dominance. The second best one is still the MOEA/D algorithm because it does not rely on Pareto-dominance. The domination in the toy problem is shown in Figure 3. Also, for an increase in the number of decision variables, we observe no change in terms of the GD and IGD values relation among the algorithms (Tables I and II), as well as visually in Figures 6, 7 and 8.

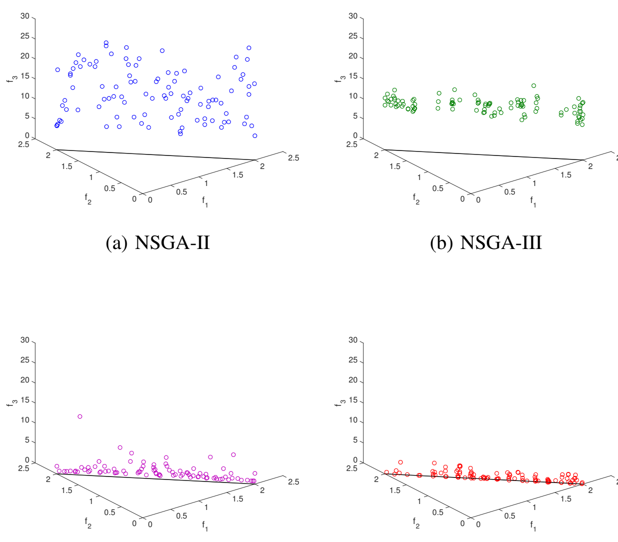
WFG3: The ϵ-MOEA is shown to work quite well for the Manhattan-based DMP and the toy problem. Now, we examine the WFG3 problem to see whether the advantages of the ϵ-



(a) NSGA-II (b) NSGA-III

(c) MOEA/D (d) ϵ-MOEA

Fig. 7. Decision space of the 2-variable toy problem with 3 objectives. The Pareto-front is given as a solid line.

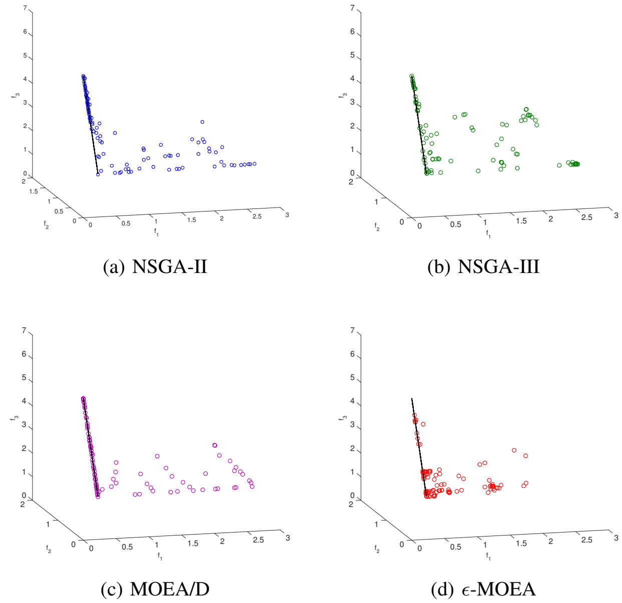


(a) NSGA-II (b) NSGA-III

(c) MOEA/D (d) ϵ-MOEA

Fig. 8. Objective space of the 100-variable toy problem with 3 objectives. The Pareto-front is given as a solid line.

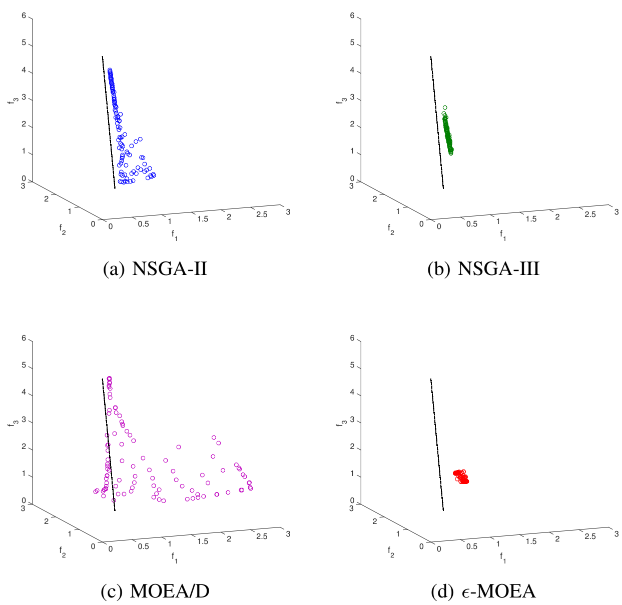
MOEA are present in another popular benchmark problem with a degenerated Pareto-front. From the GD values in Table I we see that the ϵ-MOEA performs best as long as the number of variables is kept small, but when the number of the decision variables increases, the NSGA-III algorithm takes over and delivers a closer approximation of the Pareto-front. This is shown in the Figures 9 and 10. Figure 9 shows the obtained fronts for the 6-variable problem. On average, the obtained



(a) NSGA-II (b) NSGA-III

(c) MOEA/D (d) ϵ-MOEA

Fig. 9. Objective space of the 6-variable WFG3 problem with 3 objectives. The Pareto-front is given as a solid line.



(a) NSGA-II (b) NSGA-III

(c) MOEA/D (d) ϵ-MOEA

Fig. 10. Objective space of the 120-variable WFG3 problem with 3 objectives. The Pareto-front is given as a solid line.

solutions of the ϵ-MOEA are closer to the Pareto-front than the ones of the other algorithms. However, we can see that the distribution of that fraction of solutions which are closest to the Pareto-front is much better in the other three algorithms. This is also clearly observable in Table II, where the NSGA-II, the NSGA-III and the MOEA/D algorithms obtained much smaller IGD values. The ϵ-MOEA in contrast shows some clustering of the solutions, which might be a side effect of the ϵ-grid mechanism that the algorithm uses. It is usually used to preserve diversity among the population, since in every *cell* that is also determined by the ϵ only one solution is aimed to be kept. Therefore, it seems to be counterproductive. The MOEA/D algorithm clearly shows the best IGD values for all numbers of decision variables, since its mechanism of using weight vectors is specifically designed for obtaining a good distribution among the 3 objective functions by weighting the objectives. However, this also results in a large amount of solutions found that are far away form the true Pareto-front. For larger numbers of decision variables, the best GD values are obtained by the NSGA-III algorithm. The selection based on the reference directions used in the NSGA-III seems to have a better exploitation of the search space in higher dimensions compared to the selection based on ϵ-dominance.

C. *Summary of the Analysis*

The above results indicate that ϵ-dominance implicitly helps to get rid of pseudo-optimal solutions. We observe this feature on the two problems with known Pareto-optimal sets in the decision space. In addition, the MOEA/D approach also delivers very good results. This indicates the fact that for problems with degenerated fronts and especially with hidden Pareto-optimal fronts, we should use other concepts than the pure Pareto-domination. This result is being confirmed by the experiments on WFG3 test problem with up to 30 decision variables. Nevertheless, for the WFG3 test problem with larger number of decision variables than 30, the results produced by the ϵ-MOEA are worse than the other approaches in terms of the diversity. Also the MOEA/D approach is being outperformed by the two other algorithms NSGA-II and NSGA-III in terms of Generational Distance. This is an important indication that the shape of the Pareto-optimal set, the hidden fronts and the number of the pseudo-optimal solutions are different than in the Manhattan-based DMP and the toy problem and need to be further investigated.

V. Conclusion

In this paper, we analyzed multi-objective problems with degenerated Pareto-fronts and demonstrated that there may be certain problems where the Pareto-optimal solutions are surrounded in the search space by a large set of solutions that are non-dominated to any Pareto-optimal solutions except one (Pseudo-optimal solutions). In this work, we examined how the use of ϵ-dominance can help to dominate pseudo-optimal solutions that are difficult to dominate with normal Pareto-domination. The Manhattan-metric based Distance Minimization Problem has this property, and we showed that the ϵ-MOEA, an algorithm that relies on ϵ-dominance, is able to achieve better approximations of the Pareto-front of this kind of problem. Our analysis indicates how the elimination of the need for exact domination contributes to the search, and showed that in a similar problem with these properties of solutions the ϵ-MOEA performs better also with increased dimensions of the decision space. Finally we performed experiments on the degenerated WFG3 problem, and observed that ϵ-dominance can also outperform the other used algorithms to find the degenerated front in this problem, but performs worse than the NSGA-III algorithm when the numbers of decision variables increase.

Future research might perform a closer investigation of degenerated fronts in several problems and which mechanisms can be included into search heuristics to deal with them.

REFERENCES

[1] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *Evolutionary Computation, IEEE Transactions on*, vol. 10, no. 5, pp. 477–506, 2006.

[2] H. Zille and S. Mostaghim, "Properties of scalable distance minimization problems using the manhattan metric," in *IEEE Congress on Evolutionary Computation (CEC) 2015*, May 2015, pp. 2875 – 2882.

[3] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable multi-objective optimization test problems," in *Evolutionary Computation, 2002. CEC '02. Proceedings of the 2002 Congress on*, vol. 1, 2002, pp. 825–830.

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

[5] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: solving problems with box constraints," *Evolutionary Computation, IEEE Transactions on*, vol. 18, no. 4, pp. 577–601, 2014.

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

[7] K. Deb, M. Mohan, and S. Mishra, "A fast multi-objective evolutionary algorithm for finding well-spread pareto-optimal solutions," *KanGAL report*, vol. 2003002, pp. 1–18, 2003.

[8] K. Deb, *Multi-objective optimization using evolutionary algorithms*. John Wiley & Sons, 2001, vol. 16.

[9] H. Ishibuchi, Y. Hitotsuyanagi, N. Tsukamoto, and Y. Nojima, "Many-objective test problems to visually examine the behavior of multiobjective evolution in a decision space," in *Parallel Problem Solving from Nature, PPSN XI*. Springer, 2010, pp. 91–100.

[10] H. Ishibuchi, N. Akedo, and Y. Nojima, "A many-objective test problem for visually examining diversity maintenance behavior in a decision space," in *Proceedings of the 13th annual conference on Genetic and evolutionary computation*. ACM, 2011, pp. 649–656.

[11] H. Ishibuchi, M. Yamane, N. Akedo, and Y. Nojima, "Many-objective and many-variable test problems for visual examination of multiobjective search," in *Evolutionary Computation (CEC), 2013 IEEE Congress on*, 2013, pp. 1491–1498.

[12] H. Masuda, Y. Nojima, and H. Ishibuchi, "Visual examination of the behavior of emo algorithms for many-objective optimization with many decision variables," in *Evolutionary Computation (CEC), 2014 IEEE Congress on*, July 2014, pp. 2633–2640.

[13] D. Hadka *et al.* http://www.moeaframework.org/index.html.

[14] H. Jain and K. Deb, "An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part ii: handling constraints and extending to an adaptive approach," *Evolutionary Computation, IEEE Transactions on*, vol. 18, no. 4, pp. 602–622, 2014.