

Fuzzy-Pareto-Dominance Driven Multiobjective Genetic Algorithm

Mario Köppen, Katrin Franke, Bertram Nickolay
 Fraunhofer IPK, Pascalstr. 8-9, 10587 Berlin, Germany
 {mario.koeppen|franke|nickolay}@ipk.fhg.de

Abstract

This paper presents a new approach to multiobjective optimization by evolutionary algorithm. The approach is based on fuzzification of Pareto dominance relation. Using fuzzy degrees of dominance, a set of vectors (multiple objectives) can be partially ranked. The FDD algorithm, a modification of standard genetic algorithm using this ranking scheme for the selection operations, is presented and evaluated on benchmark function.

Keywords: Multiobjective Optimization, Fuzzy Pareto Dominance, Evolutionary Algorithm.

1 Introduction and Basic Definitions

In multiobjective optimization, optimization goal is given by more than one objective to be extreme. Formally, given a domain as subset of \mathbb{R}^n , there are assigned m functions $f_1(x_1, \dots, x_n), \dots, f_m(x_1, \dots, x_n)$. Usually, there is not a single optimum but rather the so-called Pareto set of *non-dominated* solutions:

For two vectors \vec{a} and \vec{b} it is said that \vec{a} (*Pareto-dominates*) \vec{b} , when each component of \vec{a} is less or equal to the corresponding component of \vec{b} , and at least one component is smaller:

$$\vec{a} >_D \vec{b} \iff \forall i(a_i \leq b_i) \wedge \exists k(a_k < b_k). \quad (1)$$

Note that in a similar manner Pareto dominance can be related to $>$ -relation.

The subset of all vectors of a set M of vectors, which are not dominated by any other vector of M is the Pareto set (also Pareto front). The Pareto set for univariate data (single objective) contains just the maximum of the data.

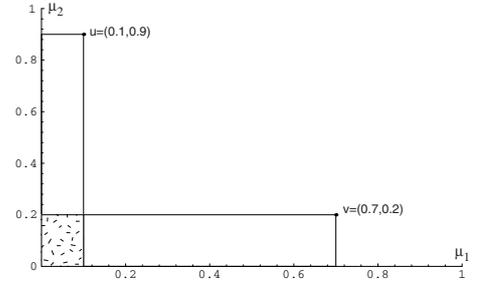


Figure 1: Definition of Fuzzy-Pareto-Dominance. Here, u dominates v by degree $0.1 \cdot 0.2 / 0.1 \cdot 0.9 = 0.2$ and is dominated by v by degree $0.1 \cdot 0.2 / 0.7 \cdot 0.2 \approx 0.143$.

We propose the fuzzification of Pareto dominance relation, given with the following definition:

It is said that vector \vec{a} *dominates* vector \vec{b} by degree μ_a with

$$\mu_a(\vec{a}, \vec{b}) = \frac{\prod_i \min(a_i, b_i)}{\prod_i a_i} \quad (2)$$

and that vector \vec{a} *is dominated by* vector \vec{b} at degree μ_p with

$$\mu_p(\vec{a}, \vec{b}) = \frac{\prod_i \min(a_i, b_i)}{\prod_i b_i} \quad (3)$$

Remarks: Note that the definitions differ in the denominator and thus are not symmetric: "dominating by degree μ " and "being dominated by degree μ " have different fuzzy values. The definition is similar to so-called subsethood degrees as introduced by Kosko [3] and has already been used for the definition of color

morphology operation [2]. For \vec{a} Pareto-dominating \vec{b} , $\mu_a(\vec{a}, \vec{b}) = 1$ and $\mu_p(\vec{b}, \vec{a}) = 1$, but $\mu_p(\vec{a}, \vec{b}) < 1$ and $\mu_a(\vec{b}, \vec{a}) < 1$.

We may use these dominance degrees to rank a set M of multivariate data (vectors) like the fitness values of a multiobjective optimization problem. Each element of M is assigned the maximum degree of being dominated by any other element of M , and the elements of M are sorted according to the ranking values in increasing order:

$$r_M(\vec{a}) = \max_{\vec{b} \in M \setminus \{\vec{a}\}} \mu_p(\vec{a}, \vec{b}) \quad (4)$$

Note that this definition is related to a set. A *ranking value* of \vec{a} within M can only be assigned with reference to a set M containing \vec{a} .

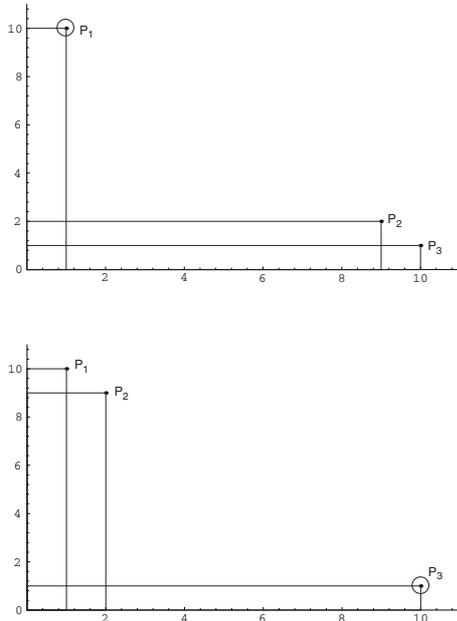


Figure 2: Counterexample: no scalar function of P components can give the same ranking as the proposed FPD ranking.

By sorting the elements of M according to the ranking values in increasing order (FPD ranking, FPD for Fuzzy-Pareto-Dominance), we obtain a partial ranking of the elements of M . For vectors with the same ranking values (like all dominated vectors), we have to assign a random ordering. There is no additional cue for complete ranking of these vectors.

An important property of this ranking scheme is that it can not be obtained by sorting of a weighted sum of

the components. More general, it can be shown that there is no scalar function of vector components of one vector at all, which will give the same ranking of the vectors of a set M . This can be shown by a simple counterexample.

Consider figure 2. We assume, that there is a scalar function f , which gives the same ranking as the FPD ordering scheme. If we take the set of three vectors $\{(1, 10), (9, 2), (10, 1)\}$, the vector with lowest ranking value is $(1, 10)$. If we take the set $\{(1, 10), (2, 9), (10, 1)\}$, the vector with lowest ranking value will be $(10, 1)$. If there would be such an f , it has to be $f(1, 10) > f(10, 1)$ from the first case, but also $f(1, 10) < f(10, 1)$ from the second case. This is a contradiction, hence there is not such a f .

In addition it should be noted that the FPD ordering is also scale-invariant.

2 FDD Algorithm

This section presents the (Fuzzy-Dominance-Driven) FDD algorithm, a Genetic Algorithm (GA) variant that employs the FPD ordering of fitness values (represented as vectors in case of multiobjective optimization) for defining selection operators. The algorithm and its components can be seen in fig. 3.

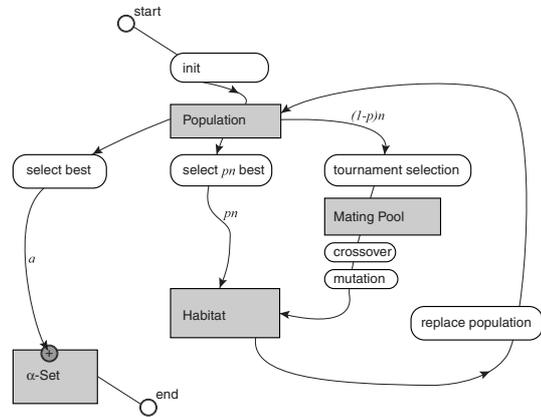


Figure 3: Schematic view of FDD algorithm.

FDD maintains four pools of individuals:

- Population: contains n individuals as in standard GA.
- Mating Pool: Contains individual pairs that were selected for crossover operation.

- Habitat: This pool is composed of individuals from other pools and used to replace the population of generation n by generation $n + 1$.
- α -Set: In this pool, all non-dominated individuals are collected. This pool also gives the output of the FDD algorithm.

After random initialization of the population, the FDD algorithm iteratively repeats the following steps until a stopping criteria (number of generations, size of α -Set) is met:

1. Rank population by FPD ordering of fitness vectors of the individuals in the population (see section 1).
2. Select *best individual* a from the ranked population (one individual with lowest ranking value) and conditionally add it to the α -set. Adding a to the α -set is only possible, when fitness of a is not dominated by the fitness of any individual already in the α -set, and if fitness of a is not equal to any individual's fitness there. In case a is added, all individuals in the α -set with fitness values dominated by fitness of a are removed from the α -set.
3. Add best pn of population individuals, according to FPD ordering ranking values, to the habitat ($0 \leq p \leq 1$).
4. Select $(1 - p)n$ pairs from population by tournament selection, using ranking values of the ranked population for tournament decision (lower ranking value counts better), and put these pairs into mating pool.
5. Apply crossover and mutation to the individuals of the mating pool, and add these newly created individuals to the habitat as well.
6. Replace population by habitat.

The FDD algorithm acquires non-dominated (with respect to their fitness values) individuals in the α -set. In an evolutionary sense, those "FDD Pareto Set" approaches the Pareto front of the multiobjective optimization problem under study.

3 Evaluation

The algorithm has been verified by using test function MOP6 from the set suggested by Coello Coello [1]. A new verification strategy will be proposed here.

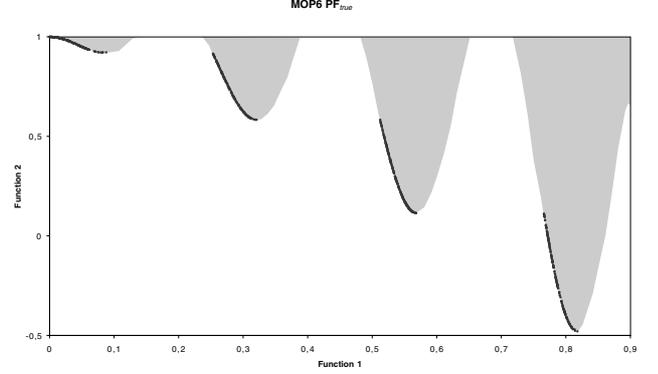


Figure 4: Location of individuals in α -set after 2000 generations of FDD algorithm for MOP6 problem.

The approach is to run FDD against random search algorithm. The performance is measured as follows:

1. Let FDD run for k generations and take the fitness values of the α -set as set M_1 .
2. Select $k \times n$ random domain values (with n the number of individuals of FDD algorithm) and compute all corresponding fitness values, giving set RM_2 .
3. Compute the Pareto set M_2 of RM_2 .
4. Compute the set M_3 of elements of M_2 that are not dominated by any element of M_1 .

The relation of $|M_1|$ to $|M_3|$ gives a measure how FDD performs against random search. We claim that this strategy is applicable to any multiobjective search algorithm on a common base.

MOP6([1], p. 111) is defined as follows:

$$F = (f_1(x, y), f_2(x, y))$$

where

$$\begin{aligned} f_1(x, y) &= x \\ f_2(x, y) &= (1 + 10y) * \\ & * \left[1 - \left[\frac{x}{1 + 10y} \right]^\alpha - \frac{x}{1 + 10y} \sin(2\pi qx) \right] \end{aligned} \quad (5)$$

with $0 \leq x, y \leq 1$ and the parameter choices $q = 4$ and $\alpha = 2$.

FDD was applied to this problem, with the following configuration:

- x and y values were encoded into bitstrings of size 40, with 20 bits for binary representation of each number.
- Population size was 50, with keeping 20 ($p = 0.4$) best from former generation in each new generation. The 20 best were selected by FPD ranking.
- Mating pool was obtained by tournament selection of 60 individuals according to FPD ranking. Two-point crossover was used, as well as bitwise one-point mutation with probability of 0.01.

Table 1: Performance of FDD against random search. M_1 is the set of non-dominated FDD individuals after n generations, M_2 the set of non-dominated individuals found by random search, and M_3 the subset of M_2 that is not dominated by any individual of M_1 . Listed are average values of set sizes after 10 FDD runs. After about 100 generations, FDD outperforms random search.

Generations	$ M_1 $	$ M_2 $	$ M_3 $
20	5.0	11.5	10.6
50	9.7	15.3	11.2
100	32.2	22.4	7.2
200	69.4	30.8	5.2
1000	411.2	122.0	0.0

Figure 4 gives the α -set fitness values of a FDD run after 2000 generations. The gray areas underlying the plot gives the range of MOP6 function values and were computed by Monte Carlo method with 5×10^7 test points. Note that fig. 4 only shows a part of the complete range of MOP6, containing the Pareto front. The α -set clearly has approached the Pareto front of the test problem.

To justify further, the α -set sizes of FDD were traced over 1000 generations in steps of 50 generations. Figure 5 shows the plot of this growing in comparison to the growing of a random search. It can be seen that in later phases, FDD produces a new non-dominated element all 2-3 generations. For checking the affordability of the FPD ordering scheme, also a plot is given

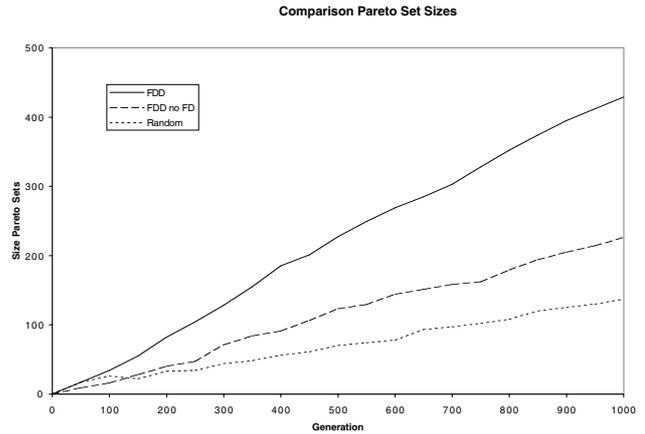


Figure 5: Growing of the number of elements in the α -set (FDD plot) against Pareto set size of random selection (Random plot) and α -set size with random instead of FPD ranking based selection of individuals (FDD no FD plot).

for FDD variant, in which selection is performed randomly. As it can be seen, α -set sizes also drops remarkably in this situation.

Finally, table 1 shows the decrease of size of set $|M_3|$ (the randomly found individuals that are not dominated by any individual in the α -set) towards 0.

References

- [1] Carlos A. Coello Coello, David A. Van Veldhuizen, Gary B. Lamont. Evolutionary Algorithms for Solving Multi-Objective Problems. Kluwer Academic Publishers, 2002.
- [2] M. Köppen, Ch. Nowack, G. Rösel. Pareto-Morphology for Color Image Processing. Proceedings SCIA99, volume 1, pages 195–202, Kangerlussuaq, Greenland, 1999.
- [3] Bart Kosko. Neural Networks and Fuzzy Systems. Prentice Hall, 1991.