Recognizing Sets in Evolutionary Multiobjective Optimization

Ewa Gajda-Zagórska

AGH University of Science and Technology, Faculty of Electrical Engineering, Automatics, IT and Electronics, Krakow, Poland

Abstract—Among Evolutionary Multiobjective Optimization Algorithms (EMOA) there are many which find only Pareto-optimal solutions. These may not be enough in case of multimodal problems and non-connected Pareto fronts, where more information about the shape of the landscape is required. We propose a Multiobjective Clustered Evolutionary Strategy (MCES) which combines a hierarchic genetic algorithm consisting of multiple populations with EMOA rank selection. In the next stage, the genetic sample is clustered to recognize regions with high density of individuals. These regions are occupied by solutions from the neighborhood of the Pareto set. We discuss genetic algorithms with heuristic and the concept of well-tuning which allows for theoretical verification of the presented strategy. Numerical results begin with one example of clustering in a single-objective benchmark problem. Afterwards, we give an illustration of the EMOA rank selection in a simple two-criteria minimization problem and provide results of the simulation of MCES for multimodal, multi-connected example. The strategy copes with multimodal problems without losing local solutions and gives better insight into the shape of the evolutionary landscape. What is more, the stability of solutions in MCES may be analyzed analytically.

Keywords— *basin of attraction, clustering, genetic algorithm, multiobjective optimization.*

1. Introduction

The aim of this paper is to present new algorithmic methods for recognizing sets and separating neighbourhoods of the Pareto sets in multiobjective problems (Multiobjective Clustered Evolutionary Strategy, MCES). We propose theoretical and experimental verification of the presented strategy.

Presented algorithmics allows to interpret the neighbourhoods of the Pareto sets like basins of attraction of the sought solutions defined for single-objective optimization problems. It also helps to separate groups of solutions when the Pareto set is non-connected. What is more, recognizing sets in multiobjective problems provides better insight into understanding the properties of the problem and the shape of the search landscape which can be helpful when further postprocessing is required (e.g., engineering problems). Another important advantage of MCES is the possibility to reduce the number of starting points for local search methods to the number of sets found. This is crucial for many-objective functions which often have an infinite number of optimal solutions. Finally, we mention difficult

multiphysics inverse problems which are extremely costly and hard to solve (see e.g. [1]).

We will focus on the idea of recognizing sets by clustering dense regions. Whereas in many papers (see i.e. [2], [3], [4]) a genetic algorithm is used as a help tool in clustering, we consider a combination of the two methods in the opposite way. Genetic algorithm here is used to provide a clustering method with the input data set. The advantages of clustering in single-objective genetic algorithms were studied by Schaefer, Adamska and Telega (CGS, see i.e. [5], [6]; well-tuning, see [7]). For other examples of two-phase global optimization strategies see [8] and [9]. Separation and estimation of the number of basins of attraction was performed by Stoean, Preuss, Stoean and Dumitrescu in [10] and in [11].

There are multiple algorithms that solve multiobjective optimization problems. The class of stochastic algorithms which approximate the Pareto set is called Evolutionary Multiobjective Optimization Algorithms (EMOA or MOEA). Usually, an EMOA aims at finding a set of Pareto-optimal solutions which may not give enough information in some cases, for example, in problems with non-connected Pareto fronts. It is difficult to extract knowledge about stability of solutions and how small perturbations affect domination among solutions from the existing algorithms. In our approach, solutions from the neighborhood of the Pareto set are detected and may be analyzed with regard to stability. For an example of analysis of stability of Pareto-optimal solutions, refer to [12]. Several examples of EMOA are presented below (for comparison see e.g. [13]).

The first method based on calculating an individual's fitness according to Pareto dominance was suggested by Goldberg in [14]. Nondominated Sorting Genetic Algorithm (NSGA) was implemented e.g. by Srinivas and Deb [15]. The selection pressure in NSGA was achieved by giving ranks determining fitness values in an iterative way: nondominated solutions are assigned rank one and temporarily removed from the population. New nondominated solutions are given rank two and so forth.

Fonseca and Fleming in [16] proposed a Pareto-based selection (*FFGA*), where an individual's rank equals the number of solutions by which it is dominated. We will refer to this type of selection later on.

In the third presented method, called Strength Pareto Evolutionary Algorithm (SPEA, see [17]), selection pressure is obtained by using an external set (archive) into which all nondominated solutions are copied in each iteration. Ranks of solutions are calculated basing on strength values of in-

dividuals stored externally. SPEA was later improved and introduced as SPEA2 in [18].

The next EMOA, by using the hypervolume measure (see e.g. [19]), maintains selection pressure as well as good distribution on the Pareto front. Hypervolume measure or *S*-metric corresponds to the size of dominated space [17]. Individuals are rated according to their contribution to the dominated hypervolume of the current population, therefore ranks are not based on relations between pairs of individuals but on relation between an individual and the whole population.

Pareto sets and fronts in multiobjective problems were investigated i.e. by Preuss, Naujoks and Rudolph in [20].

1.1. Preliminaries

We focus on global minimization problems with continuous objective functions of the form $\Phi : D \to \mathbb{R}$, $D \subset \mathbb{R}^n$, $0 \leq \Phi(x) \leq M < +\infty$, $\forall x \in D$, where D is the set of admissible solutions.

In the multiobjective optimization, we are given $k \geq 2$ objective functions

$$f_i : U \to [0,M] \subset \mathbb{R}, \ M < +\infty, \ i \in \{1,\ldots,k\} \qquad (1)$$

defined over some search space U, which might be implicitly defined by constraints. We assume the search space U to be finite $\#U = r < +\infty$ and that all objectives shall be minimized. Therefore we are interested in solving

$$\min\left\{ f(p) = (f_1(p),\ldots,f_k(p))^T \mid p \in U \right\}. \qquad (2)$$

Definition 1: (Pareto dominance) For any pair $(p,q) \in U \times U$, p is said to dominate q, denoted as $p \succ q$, if and only if

$$f(p) \leq f(q) \text{ and } \exists_{i=1,\ldots,k} f_i(p) \neq f_i(q). \qquad (3)$$

One of the possible ways to solve Eq. (2) is to find or approximate the *Pareto set* $\mathscr{P}$ being the set of non-dominated elements from U and its image $f(\mathscr{P}) \subset [0,M]^k$ called the *Pareto front*.

2. Strategy

The idea of the proposed strategy MCES of detecting neighborhoods of the Pareto sets consists of combining a genetic algorithm with a clustering method.

Among many GA we would like to extinguish those which may provide best samples for clustering. The most important property, which is held, e.g., by Simple Genetic Algorithm (SGA, for details refer to Subsection 3.3), is high selection pressure to obtain solutions in the neighborhoods of extrema. The second property is maintaining global search during computations. In case of single-population algorithms (like SGA), early convergence may eliminate global search. Such a behavior may result in losing information about parts of the Pareto front, as well as not recognizing local Pareto fronts. Therefore we propose to use an algorithm having both high selection pressure and globality, called Hierarchic Genetic Strategy (HGS, see [21]).

2.1. Genetic Engine

Hierarchic Genetic Strategy is an algorithm which produces a tree-structured set of concurrent evolutionary processes (see Fig. 1). The structure changes dynamically and the depth of the HGS tree is bounded by $m < +\infty$. In the simplest form of HGS, each process' evolution is governed by SGA.
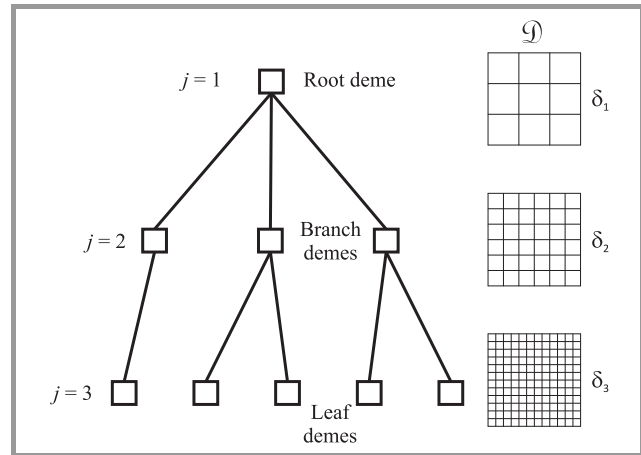

Fig. 1. HGS tree and corresponding two-dimensional meshes, $m = 3$.

HGS starts with a single root deme performing chaotic search with low accuracy. After a constant number of genetic epochs K called the *metaepoch* the root deme sprouts child-demes in the promising regions of the evolutionary landscape surrounding the best fitted individuals distinguished from the parental deme. Child-demes perform more local search with higher accuracy. The evolution in existing demes continues in the second metaepoch, after which new demes are sprouted. Demes of order m (leaves) perform local and most accurate search. The algorithm continues until the global stop condition is reached.

HGS implements two mechanisms that prevent redundancy of the search. The first one, called *conditional sprouting*, allows new demes to be sprouted only in regions which are not explored by sibling-demes (demes sprouted by the same parent). The second mechanism, called *branch reduction*, reduces demes of the same order that perform search in the common landscape region or in the regions already explored.

Different search accuracies are obtained by various encoding precisions and by manipulating the length of binary genotypes in demes at different levels. The root utilizes the shortest genotypes, while the leaves utilize the longest ones. To obtain search coherency for demes of different orders, a special kind of hierarchical nested encoding is used. Firstly, the densest mesh of phenotypes in $\mathscr{D}$ for the demes of the m-th order is defined. Afterwards, the meshes for lower order demes are recursively defined by selecting some nodes from the previous ones. The maximum diameter of the mesh δ_j associated with the demes of the order j determines the search accuracy at this level of the HGS tree (see Fig. 1). The mesh parameters satisfy $\delta_m < \ldots < \delta_1$.

Selection pressure is tightly connected with the probability of sampling measure in central parts of basins of attraction. The latter was formally proved for HGS in [21]. The theorem follows that, with certain assumptions, the sampling measures spanned by the sum of leaves in HGS are sufficiently close to the sampling measure associated with the unique fixed point of the genetic operator. Therefore, HGS is capable of detecting the same local extrema as SGA.

HGS is also more effective than SGA in finding multiple local extrema (see [22]). It consists of multiple populations which explore different areas of the search space. Even when considering only highest-order demes, the algorithm performs global search and, with a small number of individuals, can cover the whole domain.

2.2. Selection Scheme

In order to solve multiobjective optimization problems, evolution in each deme of HGS tree must be governed by an EMOA. Among EMOAs there are some selection schemes that fulfil the high selection pressure condition (several examples are described in Section 1). We will focus on selection scheme proposed by Fonseca and Fleming in [16]. Fonseca and Fleming proposed a Pareto-based selection (*FFGA*), where an individual's rank equals the number of solutions by which it is dominated. After sorting population according to the rank, fitness values are assigned to individuals by interpolating from the best (with the lowest rank) to the worst (with the highest rank) according to some function. Fitness of individuals with the same rank should be equal, so that all of them will be sampled at the same rate. We will refer to this type of selection later on, presenting a heuristic operator utilizing it.

In FFGA, selection pressure can be manipulated by using different validating functions $g \in C([0,1] \to [0,1])$ (see Subsection 3.5) which is a decreasing function transforming normalized ranks into probability distribution of the rank selection.

By applying a proper selection scheme, an EMOA converges to the Pareto front and solutions group around Pareto sets. When coupled with a multi-population strategy like HGS, an algorithm can provide a propitious sample for clustering.

2.3. Recognizing Sets

We do not restrict clustering to any particular method. Clustering here is applied to recognize regions with high density of individuals. In the presented strategy, individuals created in leaves concentrate on the neighborhoods of the Pareto-set which may be interpreted like basins of attraction of the sought solutions defined for single-objective optimization problems (see Subsection 3.1). We are aiming at finding a full-measure hull of the set of optimal solutions. Two problems appear while considering a method of recognizing sets by clustering the regions with high density of the sampling measure: What genetic algorithms should be used to provide a sample for clustering? Is it possible

to verify such a strategy theoretically? The first question is already answered – we should take advantage of algorithms combining global search with high selection pressure, e.g., used in MCES. The second question will be addressed in the following chapter.

3. Theoretical Verification

To verify the strategy theoretically, we present several concepts. Firstly, we show the theorem of clustering to recognize the basins of attraction in single-objective optimization problems. Afterwards, we move on to Simple Genetic Algorithm and the definition of genetic algorithms with heuristic. Next, a heuristic for a particular class of EMOA is presented. We finish the theoretical part with algorithms preserving the property of being well-tuned to the problem.

3.1. Basins of Attraction

We begin with necessary definitions. Let $L(y) = \{x \in D : \Phi(x) \le y\}$ and $\widehat{L}(y) = \{x \in D : \Phi(x) < y\}$ stand for two types of level sets of function Φ. $L_x(y)$ and $\widehat{L}_x(y)$ denote the connected parts of $L(y)$ and $\widehat{L}(y)$ (respectively) that contain x. For an arbitrary fixed x^* being a stationary point of function Φ let $\overline{y}(x^*) \in \mathbb{R}$ be defined as follows:

$$\overline{y}(x^*) = \begin{cases} \min \left\{ \begin{array}{l} y : \exists x^{**} \text{ isolated stationary point of } \Phi, \\ x^{**} \ne x^*, x \in L_{x^*}(y) \end{array} \right\} \\ \qquad\qquad\qquad\qquad\qquad \text{if } x^{**} \text{ exists} \\ \min_{x \in \partial D} \Phi(x) \qquad\qquad \text{otherwise}, \end{cases}$$

(4)

where $x \in \partial D$ denotes points on the boundary of the domain.

Definition 2: [5] The basin of attraction $\mathscr{B}_{x^*}$ of a local minimizer x^* is the connected part of $\widehat{L}_{x^*}(\overline{y}(x^*))$ that contains x^*.

The process of set recognition begins with a genetic sample produced by a selected genetic algorithm. The sample is divided into clusters to discover groups in the data. Formally, *clusters* are non-empty, exclusive subsets $X_1, \dots X_k$; $k \le m$ which are the results of constructiong a partition of a discrete data set $X = x_1, \dots x_m$. In the presented approach, a cluster is a discrete data set located in the basin of attraction of an isolated local minimizer x^+ of Φ.

After detecting clusters, we look for a cluster extension for each local minimizer x^+. A *cluster extension* is a closed set of positive measure which is included in $\mathscr{B}_{x^+}$ and contains x^+ in its interior. In this sense, cluster extensions approximate the basins of attraction and are located in their central parts.

Cluster extensions detection has several advantages. It allows detection and approximation of central parts of basins of attraction thus helps to determine groups of points from which local search may be started. The desired situation

is to separate local extrema to reduce the number of local searches to one in each basin of attraction of a local extremum.

What is more, the combination of genetic algorithms with clustering methods provides the possibility to analyze the stability of minimizers. Cluster extensions recognition is also useful in sequential niching strategy, to deteriorate fitness. Basins of attraction can be recognized and separated, which prevents repeated search of depressed regions of the space and repeated convergence to the same solutions. Therefore, computation time can be reduced.

The importance of set detection is even more clear in case of multiobjective optimization where we seek for a full-measure hull of Pareto-optimal solutions. This set may be interpreted as basin of attraction defined for local minimizers.

3.2. Basic Theorem of Genetic Algorithms

In this paper we consider genetic algorithms, from which the simplest operate on a single population being the multiset $P = (U, \eta)$ of the search space members called *individuals*, while U is called now *genetic universum*. A genetic universum is denoted by Ω when it is composed of all binary strings of the finite, prescribed constant length $l \in \mathbb{N}$. In this case $\Omega = \{(a_0, a_1, \ldots, a_{l-1}); a_i \in \{0, 1\}, i = 0, 1, \ldots, l-1\}$.

The occurrence function $\eta : U \to \mathbb{Z}_+ \cup \{0\}$ returns $\eta(i)$ which is the number of individuals with the genotype $i \in U$. The population cardinality is denoted by μ and $\mu = \sum_{i \in U} \eta(i) < +\infty$.

The algorithm consists in producing a sequence of populations $\{P^t\}$ in the consecutive *genetic epochs* $t = 1, 2, \ldots$ starting from the population P^0 uniformly sampled from U. The mixing and selection operations depend on the algorithm. In particular, in case of MOEA the latter is often performed with respect to the Pareto–dominance relation (see e.g. [23]).

Each finite population represented as the multiset $P = (U, \eta)$ may be identified with its frequency vector $x = \{\frac{1}{\mu} \eta(p)\}, p \in U$ and all such vectors belong to the finite subset X_μ of the Vose simplex (see e.g. [9])

$$\Lambda^r = \left\{ x = \{x_p\}; 0 \le x_p \le 1, p \in U, \sum_{p \in U} x_p = 1 \right\}. \quad (5)$$

3.3. Simple Genetic Algorithm

Simple Genetic Algorithm (introduced by Vose in [24]) applies to optimization problems with one fitness function $f : \Omega \to [0, M], M < +\infty$. It is a method to transform a population P_t to the next epoch population P_{t+1}. Both populations are multisets of binary strings from the binary genetic universum Ω of the final cardinality $r < +\infty$. Selection of two individuals x, y from population P_t is performed by multiple sampling in proportional roulette selection. An individual added to the next epoch population P_{t+1} is produced from x and y with the mixing operation (see below).

Creation of new individuals by selection and mixing is performed until P_{t+1} contains μ elements.

The proportional selection operator $F : \Lambda^r \to \Lambda^r$ is a mapping

$$F(x) = \frac{diag(f)x}{(f, x)}, \quad (6)$$

where the fitness function f is represented by the vector of its values f $(f_1, f_2, \ldots, f_r) \in \mathbb{R}^r; f_p = f(p), p \in U$ and $diag(f)$ denotes the $r \times r$ diagonal matrix with the diagonal f.

The mixing operator $M \in C^1(\Lambda^r \to \Lambda^r)$ introduced by Vose expresses the binary mutation and positional crossover

$$M(x)_p = (\sigma_p x)^T \mathbf{M} \sigma_p x, \quad \forall x \in \Lambda^r, \ p \in U, \quad (7)$$

where σ_p stands for the $r \times r$ dimension permutation matrix with the entries $(\sigma_p)_{q,k} = [q \oplus k = p], \ p, q, k \in U$. The entries $\mathbf{M}_{p,q}$ of the symmetric $r \times r$ matrix $\mathbf{M}$ express the probability of obtaining the genotype being the string of zeros from the parents $p, q \in U$ by crossover and mutation.

3.4. Algorithms with Heuristic

An important group of algorithms which properties can be theoretically verified are genetic algorithms that admit a heuristic operator. Such algorithms will be called genetic algorithms with heuristic.

Definition 3: The mapping $\mathscr{H} \in C(\Lambda^r \to \Lambda^r)$ will be called the heuristic of the particular class of genetic algorithms if:

1. $\mathscr{H}(x)$ is the expected population in the epoch that immediately follows the epoch in which the population vector $x \in \Lambda^r$ appeared,

2. H is the evolutionary law of the abstract, deterministic, infinite population algorithm (we assume that it exists in the considered class). In other words, the infinite population algorithm is the dynamic system that starts from a particular initial population $x^0 \in \Lambda^r$ and then passes consecutively by $\mathscr{H}(x^0), \mathscr{H}^2(x^0), \mathscr{H}^3(x^0), \ldots$.

3. Each coordinate $(\mathscr{H}(x))_p$ is equal to the sampling probability of the individual with the genotype $p \in U$ in the epoch that immediately follows the epoch in which the population $x \in \Lambda^r$ appears.

The heuristic operator (also called the genetic operator) for SGA is a mapping $H : \Lambda^r \to \Lambda^r$ composed of selection and mixing

$$H = M \circ F. \quad (8)$$

SGA is one of a few instances of genetic algorithms for which the probability distribution of sampling the next epoch population can be delivered explicitly (see [24]).

3.5. EMOA Heuristic

The second example of a genetic algorithm with heuristic pertains to multiobjective optimization. It was introduced in [25].

Selection operator in the presented algorithm was inspired by the Pareto-based ranking procedure FFGA described in Subsection 2.2.

Let us start with the definition of the *binary Pareto dominance matrix*

$$\Xi \in \{0,1\}^r \times \{0,1\}^r; \ \ \Xi_{p,q} = \begin{cases} 1 & \text{if } q \succ p \\ 0 & \text{otherwise.} \end{cases}, \ \forall \ p,q \in U, \tag{9}$$

which characterizes the Pareto dominance relation among the genotypes from U for the particular multiobjective optimization Eq. (2). The p-th entry of the vector $(\Xi \ \eta)$ represents the number of individuals which dominate the individual with the genotype p belonging to the population $P = (U, \eta)$ (e.g. $\eta(p) > 0$).

Next, we introduce function $\xi : \Lambda^r \to [0,1]^r$ of the form

$$\xi(x) = \Xi \ x. \tag{10}$$

The function is well defined for both finite and infinite populations. Its value $\xi(x)_p$ gives the rank of all individuals with the genotype $p \in U$ contained in the population P represented by its frequency vector x and in case of finite population of the cardinality $\mu < +\infty$ may be interpreted as the relative number of individuals that dominate the individual with the genotype p.

It is also required to introduce two following functions. A decreasing *validating function* $g \in C([0,1] \to [0,1])$ is necessary to obtain the probability distribution of the rank selection. As a simple example of a function correlated with the rank-based fitness assignment method [16] we can take $g(\zeta) = 1 - \zeta$. The second function $G : [0,1]^r \to [0,1]^r$ such that $G(x)_p = g(x_p)$, $p \in U$ is introduced for technical purposes.

The probability of selecting the individual $p \in U$ from the current EMOA population P represented by the vector $x \in \Lambda^r$ equals to

$$\Pr(p) = \frac{1}{x^T \ G(\xi(x))} \ g((\xi(x))_p) \ x_p. \tag{11}$$

Using previously introduced functions, we define the selection operator $F : \Lambda^r \to \Lambda^r$ for the EMOA rank selection

$$F(x) = \frac{1}{x^T \ G(\Xi \ x)} \ \text{diag}(x) \ G(\Xi \ x), \tag{12}$$

where $\text{diag}(x)$ denotes the $r \times r$ diagonal matrix with the diagonal x.

In each EMOA epoch, selection is followed by genetic operations (e.g., mutation, crossover) which can be represented by the mixing operator $M \in C^1(\Lambda^r \to \Lambda^r)$. No specific restrictions for this mapping are imposed. For an exemplary mixing operator see Eq. (7).

Finally, similarly like in case of SGA, we compose selection and mixing to obtain a heuristic operator of the particular class of EMOA considered in this paper

$$\mathcal{H} = M \circ F. \tag{13}$$

If the mixing operator is strictly positive, e.g., $M(x)_p > 0$, $\forall x \in \Lambda^r$, $\forall p \in U$, then the algorithm possesses the asymptotic guarantee of success, e.g., it will reach the population which contain all points lying in the Pareto set after an infinite number of epochs.

Definition 4: We say that H is *focusing* if there exists a nonempty set of fixed points $\mathcal{K} \subset \Lambda^r$ of H that for all $x \in \Lambda^r$ the sequence $\{H^t(x)\}$ converges in Λ^r to $w \in \mathcal{K}$ for $t \to +\infty$.

Theorem 1: [25] Assuming that the heuristic $\mathcal{H}$ is focusing and the mixing operator is strictly positive, the sampling measure concentrates on the set of fixed points of $\mathcal{H}$ if $\mu \to +\infty$ and $t \to +\infty$.

The theorem (for details, refer to [25]) is an extension of a similar theory introduced by Vose for SGA and has great importance in verifying MCES. Applied rank selection causes the individuals to concentrate on the neighborhood of Pareto-optimal solutions and produces a sample ready to clustering.

3.6. Well-Tuning

For genetic algorithms with heuristics it is possible to introduce a condition which is connected with the property of the frequency of solutions included in some central parts of basins of attraction being significantly higher than in other parts (see e.g. [7]).

Definition 5: [7]
A particular class of SGA with heuristic H is *well-tuned* with respect to a finite set of local minimizers $\mathcal{W}$ if:

1. H is focusing and the set of its fixed points $\mathcal{K}$ is finite,

2. $\forall x^* \in \mathcal{W} \ \exists C(x^*)$ closed set in D such that $x^* \in C(x^*) \subset \mathcal{B}_{x^*}$, $meas(C(x^*)) > 0$ and

$$\rho_w(x) \geq threshold, x \in C(x^*) \tag{14}$$

$$\rho_w(x) < threshold, x \in D \setminus \bigcup_{x^* \in \mathcal{W}} C(x^*), \tag{15}$$

where $w \in \mathcal{K}$ is a fixed point of H, ρ_w is a measure density over D corresponding to a population $w \in \Lambda^r$ and the positive constant threshold stands for the definition's parameter.

The parameter introduced by the above definition allows distinguishing whether the measure density induced by a limit population can be successfully used to separate local minimizers and to roughly locate them in the admissible set.

An important feature of algorithms well-tuned to the problem is that by increasing population size we get a higher chance of recognizing sets by cluster analysis methods.

Basing on the obtained results, we claim that Evolutionary Multiobjective Optimization Algorithm with heuristic is *well-tuned* if the fixed points of the heuristic correspond to densification of sampling measures in the neighborhoods of the Pareto set. Densification of sampling measure causes points to group around Pareto-optimal solutions where they can be recognized by clustering methods.

It was proved in [9] that if a heuristic H is focusing, the sampling measures of the algorithm converge to the measure given by the set of fixed points of H. Taking into consideration the presented EMOA heuristic with rank selection, we conclude that the level set of a particular density of the sampling measure for this selection will be the neighborhood of the Pareto set. Besides that, it is possible to asymptotically approximate that level set.

4. Experiments

We present three experimental examples. The first one refers to clustering genetic sample in a single-objective problem. The second shows rank selection in a simple multiobjective task. Finally, the third example is an application of the MCES to the benchmark problem.

4.1. Clustered Genetic Search in Single-Objective Problems

Clustering in single-objective problems was investigated by Schaefer, Adamska and Telega. The following example was presented in [5].

A two-dimensional test function

$$f(x,y) = sin(xy) + 1, \quad (x,y) \in [-3,3] \times [-3,3] \quad (16)$$

was selected to illustrate CGS abilities of coping with multimodal functions.

The objective is shown in Fig. 2. The multiple minima of the function constitute one-dimensional manifolds, which provide an additional difficulty. The authors used HGS as a genetic engine and compared two types of CGS (HC-CGS and DR-CGS, for details refer to [5]) which can be applied to solve the problem. In both cases, the algorithms found several cluster extensions which were recognized only by means of analysis of the density of individuals. They conclude that the recognized sets can be treated as central parts of basins of attractors; starting from each point of a cluster extension at least one point of the same manifold may be reached.
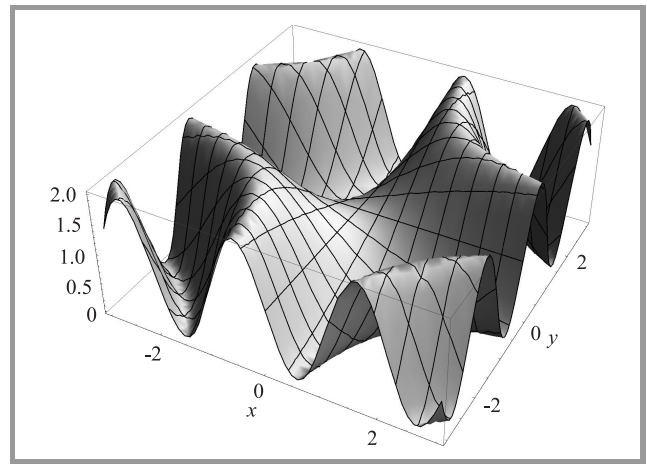

Fig. 2. The objective function f.

4.2. EMOA Rank Selection Example

The next example shows EMOA rank selection (see Eq. 12) in a two-criteria, two-dimensional minimization problem.

We take two simple objective functions with $(x,y) \in [0,4] \times [0,4]$:

$$f_1(x,y) = x + y \quad (17)$$

$$f_2(x,y) = (x-2)^2 + (y-2)^2. \quad (18)$$

We represent each individual as a binary code of length 12. The objective space is divided into a mesh of 2^{12} tiles and each tile has one representing individual corresponding to the centre of the tile. We consider a whole set of individuals and begin with computing the values of the binary Pareto dominance matrix (see Eq. (9)). Next, for each individual, we calculate rank (the number of individuals dominating it) and normalize that value. Ranks are presented as a landscape in Fig. 3. One should notice, that ranks are calculated in discrete domain but in the plot are linked for visualization purposes. For the same reasons, we focus on solutions with $rank < 0.5$.
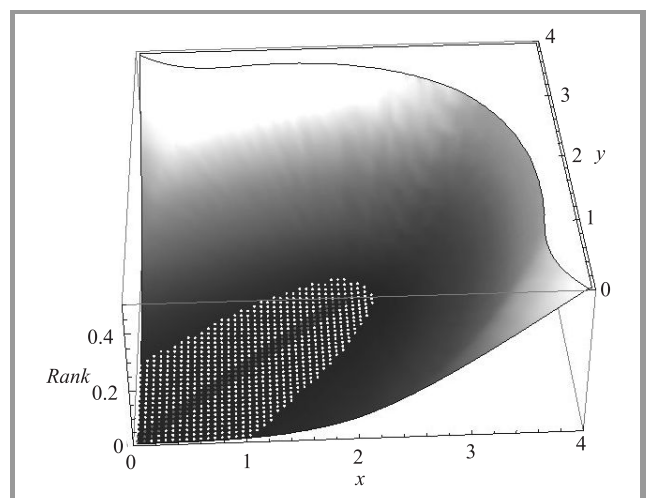

Fig. 3. Ranks of solutions in decision space. Dark grey points represent Pareto-optimal solutions, light grey points represent solutions close to optimal.

Additionally, Pareto-optimal solutions (with *rank* $= 0$) are marked dark grey and solutions being close to optimal (*rank* < 0.01) are marked light grey. Therefore, we may see a central part of the valley which may be interpreted analogically as the basin of attraction in single-objective problems.
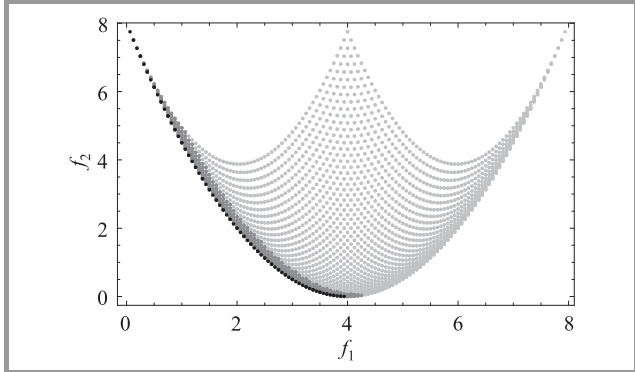

Fig. 4. Solutions in objective space. Black points represent the Pareto front, dark grey points represent solutions close to optimal.

In Fig. 4 we present solutions in the space of objectives. Pareto-optimal solutions (in this case – the Pareto front) are marked black and solutions being close to optimal (*rank* < 0.01) are marked dark grey. It is clear, that solutions from the level set of Pareto-optimal are located in the neighbourhood of the Pareto-front. What is more, concentrating of individuals on the set surrounding Pareto-optimal solutions may be used to construct a stop criterion for a particular class of EMOA.

4.3. Clustering in Multiobjective Case

As a third example we present results of a simulation of MCES combining HGS engine with EMOA rank selection and clustering.

For a case study we have chosen a two-criteria, two-dimensional minimization problem with the following objective functions:

$$f_1(x,y) = x \qquad (19)$$

$$f_2(x,y) = g(y)\left(1 - \sqrt{\frac{x}{g(y)}} - \frac{x}{g(y)}\sin(10\pi x)\right), \qquad (20)$$

where $g(y) = 1 + 9y$, $(x,y) \in [0,1] \times [0,1]$ (see Fig. 5).

The problem is quite difficult to solve because it is multimodal and its Pareto-optimal front consists of several non-connected parts.

As a genetic engine in the example we use a two-level HGS with rank selection presented in the paper. Root deme consists of 50 individuals and the mutation probability is 0.05. The stop condition is fulfilled when the root population finishes the 20th metaepoch. After each metaepoch leaves are sprouted in the best places found by root (around individuals with lowest ranks). Each leaf population consists of 10 individuals and the mutation probability is 0.005. We have limited leaf evolution to 5 metaepochs.
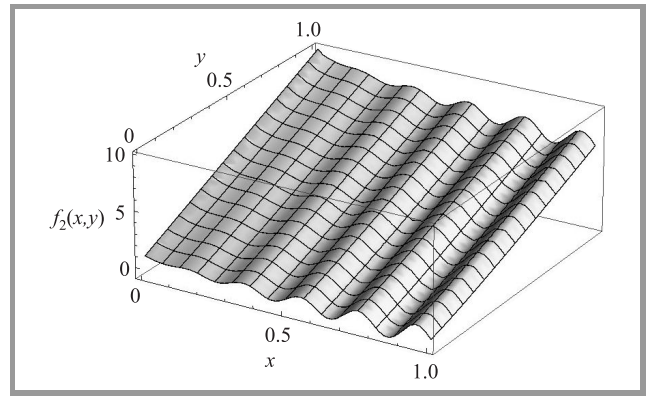

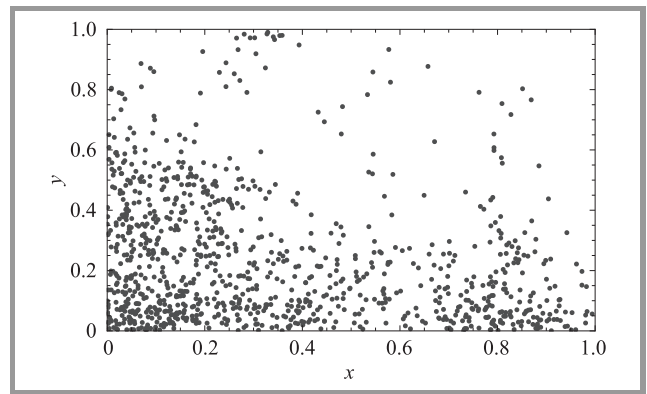**Fig. 5.** Objective function f_2 (see Eq. (20)).

Fig. 6. Root individuals in the decision space.

In Fig. 6 we present all individuals created by root. The individuals are quite well-spread in the entire search space and group in the regions which contain solutions with low ranks. The same individuals are presented in Fig. 7 in the objective space. Recognized parts of the Pareto front are visible in the lower part of the plot.
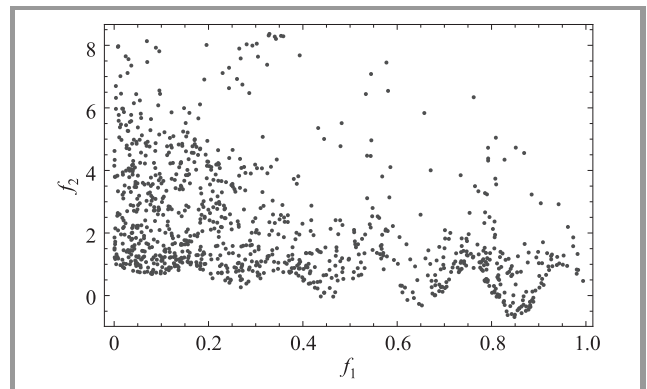

Fig. 7. Root individuals in the objective space.

Leaves continue exploration in most interesting parts of the landscape. Most of these regions are the neighborhoods of the Pareto-optimal sets (see Fig. 8).

Afterwards, the results of search in leaves are being clustered my k-medoids method (see, i.e., [26]). In the pre-
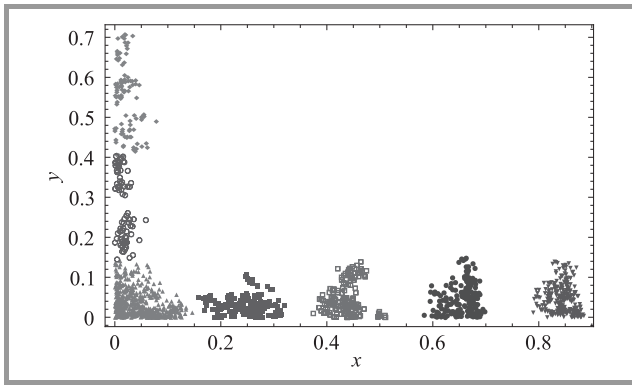
Fig. 8. Individuals created in leaves, decision space.

sented example problem, found clusters represent existing parts of Pareto front very well. Two upper clusters (Fig. 9) are the results of early sprouting in regions interesting at the beginning of computation in root whereas the remaining ones are exactly the solutions we were looking for.
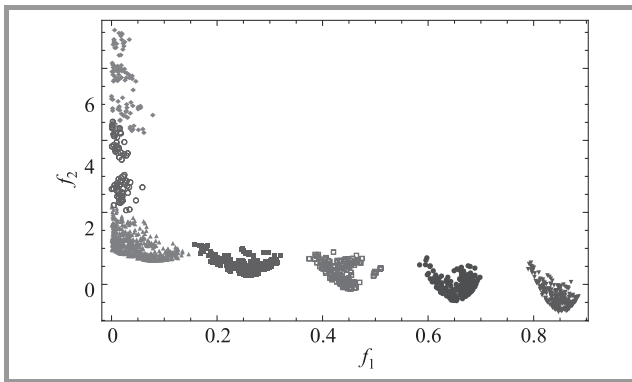

Fig. 9. Individuals created in leaves, objective space.

To conclude, in this chapter we presented examples which show the strategy in practice. It may be successfully applied to multimodal problems and gives a better insight into the shape of problem landscape. Clustering results of genetic search allows detecting basins of attractions of solutions in single-objective optimization tasks as well as analogical sets of individuals in neighborhoods of Pareto-optimal solutions in multiobjective case.

5. Conclusions and Future Research

- The presented strategy of solving a Pareto optimization problem gives additional knowledge about the shape of the evolutionary landscape. What is more, it copes with multimodal problems without losing local solutions.

- Set recognition allows for detecting central regions of the basins of attraction and, as a result, starting points for local search methods can be limited to one in each basin of attraction.

- MCES can be partially theoretically verified by using concepts of EMOA heuristic and well-tuning.

- We suppose that presented methods can be applied to solve multiobjective inverse problems in cooperation with hp-adaptive direct problem solving methods.

- In future papers, we plan to develop the theorem allowing for verification of the strategy and investigate the property of well-tuning of EMOA.

Acknowledgements

References

[1] B. Barabasz, S. Migórski, R. Schaefer, and M. Paszyński, "Multi-deme, twin adaptive strategy hp-hgs", *Inverse Problems in Sci. Engin.*, vol. 19, no. 1, pp. 3–16, 2011.

[2] M. C. Cowgill, R. J. Harvey, and L. T. Watson, "A genetic algorithm approach to cluster analysis", *Comput. Mathem. Appl.*, vol. 37, no. 7, pp. 99–108, 1999.

[3] E. Hruschka and N. Ebecken, "A genetic algorithm for cluster analysis", *Intell. Data Analysis*, vol. 7, no. 15, pp. 15–25, 2003.

[4] U. Maulik and S. Bandyopadhyay, "Genetic algorithm-based clustering technique", *Pattern Recognition*, vol. 33, no. 9, pp. 1455–1465, 2000.

[5] R. Schaefer, K. Adamska, and H. Telega, "Clustered genetic search in continuous landscape exploration", *Engin. Appl. Artificial Intell.*, vol. 17, no. 4, pp. 407–416, 2004.

[6] K. Adamska, "Genetic clustering as a parallel algorithm for approximating basins of attraction", in *Parallel Processing and Applied Mathematics*, R. Wyrzykowski, J. Dongarra, M. Paprzycki, and J. Wasniewski, Eds., vol. 3019, LNCS. Berlin/Heidelberg: Springer, 2004, pp. 536–543.

[7] R. Schaefer and K. Adamska, "Well-tuned genetic algorithm and its advantage in detecting basins of attraction", in *Proc. 7th Conf. Evol. Algorithms Global Optimiz.*, Kazimierz Dolny, Poland, 2004, pp. 149–154.

[8] *Handbook of Global Optimization vol. 2*. P. M. Pardalos and H. E. Romeijn, Eds. Kluwer, 1995.
Kluwer, 1995.

[9] R. Schaefer and H. Telega, *Foundation of Global Genetic Optimization*. Springer, 2007.

[10] C. Stoean, M. Preuss, R. Stoean, and D. Dumitrescu, "Ea-powered basin number estimation by means of preservation and exploration", in *Parallel Problem Solving from Nature – PPSN X*, G. Rudolph, T. Jansen, S. M. Lucas, C. Poloni, and N. Beume, Eds., vol. 5199, LNCS, Berlin: Springer, 2008, pp. 569–578.

[11] C. Stoean R. Stoean, M. Preuss, "Approximating the number of attraction basins of a function by means of clustering and evolutionary algorithms", in *8th Int. Conf. Artif. Intelli. Digit. Commun. AIDC 2008*, *Research Notes in Artificial Intelligence and Digital Communications*, N. Tandareanu, Ed., Reprograph Press, 2008, pp. 171–180.

[12] V. A. Emelichev and E. E. Gurevsky, "On stability of a pareto-optimal solution under perturbations of the parameters for a multicriteria combinatorial partition problem" [Online]. Available: http://www.math.md/publications/csjm/issues/v16-n2/8827/

[13] E. Zitzler, "Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications", Ph.D. thesis, ETH Zurich, Switzerland, 1999.

[14] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learnings*. Reading, Massachusetts: Addison-Wesley, 1989.

[15] N. Srinivas and K. Deb, "Multiobjective optimization using nondominated sorting in genetic algorithms", *Evolutionary Comput.*, 1994.

[16] C. M. Fonseca and P. J. Fleming, Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. *Genetic Algorithms: Proceedings of the Fifth International Conference, pp. 416-423*, 1993.

[17] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength pareto evolutionary algorithm", *IEEE Trans. Evolutionary Comput.*, vol. 3, no. 4, pp. 257–271, 1999.

[18] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization", in *Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems (EUROGEN 2001)*, International Center for Numerical Methods in Engineering (CIMNE), 2002, pp. 95–100.

[19] M. Emmerich, N. Beume, and B. Naujoks, "An EMO algorithm using the hypervolume measure as selection criterion", in *Evolutionary Multi-Criterion Optimization: Third International Conference EMO 2005*, Springer, 2005, pp. 62–76.

[20] M. Preuss, B. Naujoks, and G. Rudolph, "Pareto set and EMOA behavior for simple multimodal multiobjective functions", in *Parallel Problem Solving from Nature – PPSN IX*, T. P. Runarsson, H.-G. Beyer, E. Burke, J. J. Merelo-Guervós, L. D. Whitley, and X. Yao, Eds., vol. 4193, LNCS, Berlin: Springer, 2006, pp. 513–522.

[21] J. Kolodziej, "Modelling hierarchical genetic strategy as a family of markov chains", in *Proc. Int. Conf. Parallel Proces. Applied Mathem.-Revised Papers*, Springer, 2002, pp. 595–598.

[22] R. Schaefer and J. Kolodziej, "Genetic search reinforced by the population hierarchy", in *Foundations of Genetic Algorithms 7*, R. Poli, K. A. De Jong, and J. E. Rowe, Eds., Morgan Kaufman, 2003, pp. 383–388.

[23] C. A. Coello Coello and G. B. Lamont, *Applications of Multiobjective Evolutionary Algorithms*. World Scientific, 2004.

[24] M. D. Vose, *The Simple Genetic Algorithm*. MIT Press, 1999.

[25] E. Gajda, R. Schaefer, and M. Smolka, "Evolutionary multiobjective optimization algorithm as a markov system", in *Parallel Problem Solving from Nature – PPSN XI*, Springer, 2010, pp. 617–626.

[26] H. Park and C. Jun, "A simple and fast algorithm for k-medoids clustering", *Expert Systems with Applications*, vol. 36, no. 2, part 2, pp. 3336–3341, 2009.

Ewa Gajda-Zagórska received the M.Sc. degree in Computer Science at Jagiellonian University, Kraków, Poland, in 2009. Since then, she has been a Ph.D. student of Computer Science at AGH University of Science and Technology, Kraków, Poland. Her research interests are evolutionary algorithms and multiobjective optimization.

E-mail: gajda@agh.edu.pl
AGH University of Science and Technology
Faculty of Electrical Engineering, Automatics,
IT and Electronics
A. Mickiewicza Av. 30
B-1 building
30-059 Krakow, Poland