Scientific Research Publishing

Generating Epsilon-Efficient Solutions in Multiobjective Optimization by Genetic Algorithm

El-Desouky Rahmo[1,2], Marcin Studniarski[3]

[1]Department of Mathematics, Faculty of Science, Taif University, Khurma, KSA
[2]Mathematics Department, Faculty of Science, Mansoura University, Mansoura, Egypt
[3]Faculty of Mathematics and Computer Science, University of Łódź, Łódź, Poland
Email: desoukyr@hotmail.com

Abstract

We develop a new evolutionary method of generating epsilon-efficient solutions of a continuous multiobjective programming problem. This is achieved by discretizing the problem and then using a genetic algorithm with some derived probabilistic stopping criteria to obtain all minimal solutions for the discretized problem. We prove that these minimal solutions are the epsilon-optimal solutions to the original problem. We also present some computational examples illustrating the efficiency of our method.

1. Introduction

The goal of multiobjective optimization, also called vector optimization, is to find a certain set of optimal (efficient) elements of a nonempty subset of a partially ordered linear space. However, finding an exact description of this set often turns out to be practically impossible or computationally too expensive. Therefore, many researchers have focused their efforts on approximation procedures and approximate solutions (see e.g. [1] [2] and references therein).

More than three decades ago, the notion of ε-efficiency has been introduced by Loridan [3] for multi-objective optimization problems (MOPs). Afterwards, this concept has been used e.g. in [2] [4] [5]. To deal with a continuous multi-objective optimization problem, one has to consider a finite discretization of the set of feasible points (see Section 3 below). The discretization of the search space

is one of the efficient techniques to obtain approximate solutions for MOPs, (e.g. [6] [7]). The aim of the present paper is to develop a method of generating ε-efficient solutions (as defined in [4]) of a continuous MOP. This is achieved by discretizing the problem and then using a genetic algorithm according to the scheme described in [8]. In this way, some probabilistic stopping criteria are obtained for this procedure. They are given in the form of an upper bound for the number of iterations necessary to ensure finding all minimal elements of a finite partially ordered set with a prescribed probability. Supporting theoretical results are established and some computational examples are provided.

2. Stopping Criteria for Genetic Algorithms

In this section we review the results of [8] on probabilistic stopping criteria which will be applied later in Section 4 to some continuous multiobjective optimization problem.

2.1. Random Heuristic Search

The RHS (*Random Heuristic Search*) algorithm, described in [9], is defined by a fixed *initial population* $\hat{p}$ and a *transition rule* τ which, for a given population p, determines a new population $\tau(p)$. Iterating τ, we obtain a sequence of populations:

$$\hat{p},\ \tau(\hat{p}),\ \tau^2(\hat{p}), \cdots \tag{1}$$

Each population consists of a finite number of *individuals* which are elements of a given finite set Ω called the *search space*. Populations are *multisets*, which means that the same individual may appear more than once in a given population.

To simplify the notation, it is convenient to identify Ω with a subset of integers: $\Omega = \{0, 1, \cdots, n-1\}$. The number n is called the *size of search space*. Then a population can be represented as an *incidence vector* (see [10], p. 141):

$$v = (v_0, v_1, \cdots, v_{n-1})^{\mathrm{T}}, \tag{2}$$

where v_i is the number of copies of individual $i \in \Omega$ in the population ($v_i = 0$ if the i-th individual does not appear in the population). The *size of population* v is the number

$$r = \sum_{i=0}^{n-1} v_i. \tag{3}$$

We assume that all the populations appearing in sequence (1) have the same size r. Dividing each component of incidence vector (2) by r, we obtain the *population vector*

$$p = (p_0, p_1, \cdots, p_{n-1})^{\mathrm{T}}, \tag{4}$$

where $p_i = v_i / r$ is the proportion of individual $i \in \Omega$ in the population. In this way, we obtain a more general representation of the population which is independent of population size. It follows that each vector p of this type be-

longs to the set

$$\Lambda := \left\{ x \in \mathbb{R}^n : x_i \geq 0 \, (\forall i), \sum_{i=0}^{n-1} x_i = 1 \right\}, \tag{5}$$

which is a simplex in $\mathbb{R}^n$. However, not all points of this simplex correspond to finite populations. For a fixed $r \in \mathbb{N}$, the following subset of Λ consists of all populations of size r (see [9], p. 7):

$$\Lambda_r := \frac{1}{r} \left\{ x \in \mathbb{R}^n : x_i \in \mathbb{N} \cup \{0\} \, (\forall i), \sum_{i=0}^{n-1} x_i = r \right\}. \tag{6}$$

We now define the mapping

$$\mathcal{G} : \Lambda \to \Lambda,$$

called *heuristic* ([9], p. 9) or *generational operator* ([10], p. 144), in the following way: for a vector $p \in \Lambda$ representing the current population, $\mathcal{G}(p)$ is the probability distribution that is sampled independently r times (with replacement) to produce the next population after p. For each of these r choices, the probability of selecting an individual $i \in \Omega$ is equal to $\mathcal{G}(p)_i$, the i-th component of $\mathcal{G}(p)$.

A transition rule τ is called *admissible* if it is a composition of a heuristic $\mathcal{G}$ with drawing a sample in the way described above. Symbolically,

$$\tau(p) = \text{sample}(\mathcal{G}(p)), \ \forall p \in \Lambda. \tag{7}$$

Of course, a transition rule defined this way is nondeterministic, *i.e.*, by applying it repeatedly to the same vector p, we can obtain different results. It should also be noted that, although $\mathcal{G}(p)$ may not belong to Λ_r, the result of drawing an r-element sample is always a population of size r; therefore, it follows from (7) that $\tau(p) \in \Lambda_r$.

2.2. The Case of a Genetic Algorithm

In this subsection we consider a genetic algorithm as a particular case of the RHS. We assume that a single iteration of the genetic algorithm produces the next population from the current population as follows:

1) Choose two parents from the current population by using a selection method which can be described by some heuristic (see [9], 4.2).

2) Crossover the two parents to obtain a child.

3) Mutate the child.

4) Put the mutated child into the next population.

5) If the next population contains less than r members, return to step 1.

The only difference between the iteration described above and the iteration of the Simple Genetic Algorithm defined in ([9], p. 44) is that in our version mutation is done after crossover.

To derive our stopping criteria, we will use some properties of mutation which is generally understood as changing one element of the search space to another, with a certain probability. The way of implementing selection and crossover is not important for our model, so we omit the discussion of them (we

refer the reader to ([10], Chapter 5)). The only requirement is that the composition of the three operations (selection, crossover, mutation) can be described in terms of some heuristic.

We assume that mutation consists in replacing a given individual from Ω by another individual, with a prescribed probability. Let us denote by $u_{i,j}$ the probability that individual i mutates to j. In this way, we obtain a $n \times n$ matrix $U = \left[u_{i,j} \right]_{i,j \in \Omega}$. We denote by $\Pr(q \mid p) = \Pr(\tau(p) = q)$ the probability of obtaining a population q in the current iteration of the RHS algorithm provided the previous population is p, and by $\Pr([j] \mid p) = \mathcal{G}(p)_j$ the probability of selecting an individual $j \in \Omega$ by a single sampling of the probability distribution $\mathcal{G}(p)$. In particular, the probability of generating individual j from population p by successive application of selection, crossover and mutation is equal to (see [8], formula (7))

$$\mathcal{G}(p)_j = \Pr([j] \mid p)_{\text{scm}} = \sum_{i=0}^{n-1} u_{i,j} \Pr([j] \mid p)_{\text{sc}}, \tag{8}$$

where the subscript sc means that we are dealing with the composition of selection and crossover, and the subscript scm indicates the composition of selection, crossover and mutation. To get a whole new population, one should draw an *r*-element sample from the probability distribution (8). The probability of generating a population q in this way is then equal to (see [8], formula (8))

$$\Pr(q \mid p)_{\text{scm}} = r! \prod_{j=0}^{n-1} \frac{\left(\Pr([j] \mid p)_{\text{scm}} \right)^{rq_j}}{(rq_j)!}. \tag{9}$$

2.3. Stopping Criteria for Finding All Minimal Elements of Ω

We now consider the following multiobjective optimization problem. Let Ω be a finite search space defined in Subsection 2.1, and let $f : \Omega \to F$ be a function being minimized, where $F = \{ f(\omega) : \omega \in \Omega \}$ and $(F, \preceq)$ is a partially ordered set. An element $x^* \in F$ is called a *minimal element* of $(F, \preceq)$ if there is no $x \in F$ such that $x \prec x^*$, where the relation $\prec$ is defined by

$$(x \prec y) :\Leftrightarrow (x \preceq y \wedge x \neq y). \tag{10}$$

The set of all minimal elements of F is denoted by $\text{Min}(F, \preceq)$. We define the set of optimal solutions in our multiobjective problem as follows:

$$\Omega^* = \text{Min}_f(\Omega, \preceq) := \{ \omega \in \Omega : f(\omega) \in \text{Min}(f(\Omega), \preceq) \}. \tag{11}$$

In particular, if F is a finite subset of the Euclidean space $\mathbb{R}^k$, and $f = (f_1, \cdots, f_k)$, where each component of f is being minimized independently, then the relation $\preceq$ in F can be defined by

$$(x \preceq y) :\Leftrightarrow (x_i \leq y_i, i = 1, \cdots, k). \tag{12}$$

In this case, Ω^* is the set of all *Pareto optimal solutions* of the respective multiobjective optimization problem.

In this section, we assume that the goal of RHS is to find all elements of Ω^*.

Suppose that the set Ω^* of minimal solutions has the following form:

$$\Omega^* = \left\{ j_1, j_2, \cdots, j_m \right\}, \tag{12}$$

where the (possibly unknown) number m of these solutions is bounded from above by some known positive integer M. We say that *all elements of Ω^* have been found in the first t iterations* if, for each $l \in \{1, \cdots, m\}$, there exists $s \in \{1, \cdots, t\}$ such that $\tau^s(\hat{p})_{j_l} > 0$. This means that each minimal solution is a member of some population generated in the first t iterations.

Theorem 1 ([8], *Thm.* 6.1) *We consider the general model of genetic algorithm, described in Subsection* 2.2, *being a special case of the RHS algorithm with the heuristic $\mathcal{G}$ given by* (8). *Suppose that there exists a number $\beta \in (0,1)$ satisfying*

$$u_{i,j} \geq \beta, \quad \forall i \in \Omega, j \in \Omega^*. \tag{13}$$

Let M and t be two positive integers satisfying the inequality

$$M (1 - \beta)^{rt} < 1. \tag{14}$$

Let Ω^* be of the form (12) *with $m \leq M$. Then the probability of finding all elements of Ω^* in the first t iterations is at least*

$$1 - M (1 - \beta)^{rt}. \tag{15}$$

Corollary 2 ([8], *Cor.* 2) *We consider the same model of algorithm as in Theorem* 1. *Suppose that condition* (13) *holds for some $\beta \in (0,1)$. Let M be a given upper bound for the cardinality of Ω^*. For any $\delta \in (0,1)$, we denote by $t^*_{\min}(\delta)$ the smallest number of iterations required to guarantee that all elements of Ω^* have been found with probability δ. Then*

$$t^*_{\min}(\delta) \leq \left\lceil \frac{\ln(1-\delta) - \ln M}{r \ln(1-\beta)} \right\rceil, \tag{16}$$

where $\lceil x \rceil$ is the smallest integer greater than or equal to x.

2.4. Construction of the Set of Minimal Elements

The results of Section 2.3 give no practical way of constructing the set Ω^*. Different elements of this set are members of different populations generated by the genetic algorithm, and cannot be easily identified. To give an effective way of constructing Ω^*, some modification of the RHS is necessary.

The algorithm presented below is a combination of the RHS and the base VV (van Veldhuizen) algorithm described in ([11], 3.1). Suppose we have some RHS satisfying the assumptions of Theorem 1. It generates a sequence (1) of populations, all of them being members of Λ_r. For each $p \in \Lambda_r$, we define the set of individuals represented in population p:

$$\text{set}(p) := \left\{ \omega \in \Omega : p_\omega \neq 0 \right\}. \tag{17}$$

Then we construct a sequence $\{D_t\}$ of subsets of Ω as follows:

$$D_t := \text{set}(\tau^t(\hat{p})), \ t = 0, 1, \cdots, \tag{18}$$

where $\tau^0 := id$ is the identity mapping. Finally, we define another sequence $\{E_t\}$ of sets recursively by

$$E_0 := \text{Min}_f(D_0, \preceq), \tag{19}$$

$$E_{t+1} := \text{Min}_f(E_t \cup D_{t+1}, \preceq), \quad t = 0, 1, \cdots, \tag{20}$$

where we have used the notation Min_f as in (11). Formulas (19) and (20) define the VV algorithm.

It is shown in ([11], Prop. 1) that the sets $f(E_t)$ converge with probability 1 to $\text{Min}(F, \preceq)$ in the sense of some metric. However, as the authors comment, "The size of the sets E_t will finally grow to the size of the set of minimal elements. Since this size may be huge, this base algorithm offers only limited usefulness in practice". In fact, our considerations can have practical value only if the cardinality of Ω^* is relatively small. For continuous multiobjective optimization problems, such situation can be achieved by choosing a suitable discretization.

Our final result is the following theorem which shows that, with a prescribed probability, the sets E_t constructed by the VV algorithm are equal to Ω^* for t sufficiently large.

Theorem 3 ([8], *Thm. 7.1*) *Let the assumptions of Corollary* 2 *be satisfied. Then, with probability* δ, *we have*

$$\Omega^* = E_t, \quad \forall t \geq t_{\min}^*(\delta). \tag{21}$$

3. Generation of ε-Efficient Solutions for a Continuous Problem

Let $f : X \to \mathbb{R}^l$ be a given mapping, where X is a closed and bounded subset of $\mathbb{R}^k$. We consider the following multiobjective optimization problem:

$$\min\{f(x) : x \in X\}. \tag{22}$$

To solve this problem means to find all Pareto optimal (efficient) points of X with respect to the partial order relation in $\mathbb{R}^l$ defined by

$$(u \preceq v) :\Leftrightarrow (u_i \leq v_i, i = 1, \cdots, l). \tag{23}$$

However, in practical situations this can be very difficult or even impossible. Therefore, we shall consider a discretized version of problem (22).

For any given $\eta > 0$, we say that a subset Ω of $\mathbb{R}^k$ is a η-*discretization* of X if

$$\Omega \subset X \text{ and } X \subset \bigcup_{z \in \Omega} B(z, \eta), \tag{24}$$

where $B(x, \eta) := \{y \in \mathbb{R}^k : \|y - x\| < \eta\}$. Since X is compact, we can always find a finite η-discretization of X. The discretized multiobjective optimization problem can now be formulated as follows:

$$\min\{f(x) : x \in \Omega\}, \tag{25}$$

where the same relation (23) is considered, but now in the finite set $f(\Omega)$.

It is natural to ask whether an exact solution of problem (25) yields some sort of approximate solution of problem (22). One of the cases where a positive answer can be given is described in the proposition below. Before formulating it, we must define ε-efficient solutions, following ([4], Definition 2.3 (ii)).

Let $\varepsilon = (\varepsilon_1, \cdots, \varepsilon_l) \in \mathbb{R}^l$ be such that $\varepsilon_i > 0$ $(i = 1, \cdots, l)$. We say that a point $\overline{x} \in X$ is an ε-*efficient solution* of problem (22) if there is no $x \in X$ such that

$$f(x) \prec f(\overline{x}) - \varepsilon, \tag{26}$$

where the relation "$\prec$" is defined by formula (10).

Proposition 4 *Let* $f = (f_1, \cdots, f_l) : X \to \mathbb{R}^l$ *where* X *is compact and each function* f_i *is Lipschitz continuous with constant* $K_i > 0$ $(i = 1, \cdots, l)$. *Let* $\varepsilon \in \mathbb{R}^l$ *be such that* $\varepsilon_i > 0$ $(i = 1, \cdots, l)$. *Define*

$$\eta := \min \left\{ \frac{\varepsilon_i}{K_i} : i = 1, \cdots, l \right\}, \tag{27}$$

and let Ω *be a* η-*discretization of* X. *Denote by* Ω^* *the set of all Pareto optimal solutions of problem* (25) *(i.e.,* $\Omega^* = \mathrm{Min}_f (\Omega, \preceq)$ *as in formula* (11)). *Then every point* $\overline{x} \in \Omega^*$ *is an* ε-*efficient solution of problem* (22).

Proof. Let $\overline{x} \in \Omega^*$. Suppose to the contrary that $\overline{x}$ is not an ε-efficient solution of (22). Then there exists $x \in X$ such that (26) holds. In particular, we have

$$f_i(x) \leq f_i(\overline{x}) - \varepsilon_i, \text{ for all } i \in \{1, \cdots, l\}. \tag{28}$$

By the second inclusion in (24), there exists $z \in \Omega$ such that $\|z - x\| < \eta$. Therefore, using (27) and (28), we obtain, for all $i \in \{1, \cdots, l\}$,

$$\begin{aligned}
f_i(z) &\leq f_i(x) + |f_i(z) - f_i(x)| \\
&\leq f_i(x) + K_i \|z - x\| \\
&< f_i(x) + K_i \eta \\
&\leq f_i(\overline{x}) - \varepsilon_i + K_i \eta \leq f_i(\overline{x}),
\end{aligned}$$

which contradicts the assumption that $\overline{x} \in \Omega^*$. $\blacksquare$

4. The Main Algorithm

Consider the multiobjective optimization problem (22), where the constraint set X is a box defined by

$$X := \prod_{i=1}^{k} [\alpha_i, \beta_i], \tag{29}$$

where $\alpha_i < \beta_i$ $(i = 1, \cdots, k)$. Suppose that the assumptions of Proposition 4 are satisfied. We want to specify a η-discretization of X. We construct the set Ω as follows:

$$\Omega := \left\{ x \in \mathbb{R}^k : x_i = \alpha_i + \frac{t_i}{k_i} (\beta_i - \alpha_i), t_i = 0, 1, \cdots, k_i, i = 1, \cdots, k \right\}, \tag{30}$$

where k_i $(i = 1, \cdots k)$ are given positive integers.

Proposition 5 *For every given* $\eta > 0$, *it is possible to find the numbers* k_i *so large that the set* Ω *defined by* (30) *is a* η *-discretization of* X.

Proof. The inclusion $\Omega \subset X$ is obvious. We now prove the second inclusion in (24). For simplicity, we consider the l_∞ norm in $\mathbb{R}^k$:

$$\|x\|_\infty := \max_{1 \le i \le k} |x_i|. \tag{31}$$

Let us choose k_i so that

$$\frac{1}{k_i}(\beta_i - \alpha_i) < 2\eta. \tag{32}$$

Take any $x \in X$. Then, for each $i \in \{1, \cdots, k\}$, there exists $s_i \in \{0, 1, \cdots, k_i\}$ such that the number z_i defined by

$$z_i := \alpha_i + \frac{s_i}{k_i}(\beta_i - \alpha_i) \tag{33}$$

satisfies

$$|x_i - z_i| \le \frac{1}{2k_i}(\beta_i - \alpha_i) < \eta.$$

Then the vector $z := (z_1, \cdots, z_k) \in \Omega$ is such that

$$\|x - z\|_\infty := \max_{1 \le i \le k} |x_i - z_i| < \eta,$$

which completes the proof. ■

In the sequel we consider the following simple evolutionary algorithm which is a special case of the algorithm described in Subsection 2.4. It does not contain selection and crossover. The mutation process is very simple and consists in replacing the current population by another randomly chosen population of the same size. However, the stopping criteria described above still hold for this algorithm because their proofs make use of the properties of the mutation alone.

Algorithm 1 Parameters: $\delta > 0$ (for the stopping criterion), $\varepsilon \in \mathbb{R}^l$ (for defining η-discretization).

1) Set $t := 0$.

2) Choose randomly a population D_0 consisting of r elements of Ω.

3) Construct the set E_0 by formula (19).

4) Mutate the population D_t by replacing it by another randomly chosen population D_{t+1} consisting of r elements of Ω.

5) Construct the set E_{t+1} by formula (20).

6) If $t + 1 \ge t_{\min}^*(\delta)$, then stop and define $\bar{\Omega} := E_{t+1}$.

7) Increase t by 1 and go to Step 4.

Proposition 6 *After stopping Algorithm* 1, *the equality* $\bar{\Omega} = \Omega^*$ *holds with probability* δ, *and consequently,* $\bar{\Omega}$ *consists entirely of* ε *-efficient solutions of problem* (22) *with probability* δ.

Proof. Apply Theorem 3 and Corollary 2 with $M := \text{card}\,\Omega$ and $\beta := 1/M$ (we assume the equal probability $1/M$ of mutating i to j for every $i, j \in \Omega$). ■

5. Computational Examples

Below we report on testing the algorithm described above on some examples taken from literature. To find the set of minimal elements (*i.e.*, nondominated elements) of finite sets in Steps 3 and 5, we have used the simple algorithm for classifying a population according to non-domination, see Section 4.3.1 of [12].

Example 7 (Problem SCH in Table I of [13])

$$\min\left(f_1(x), f_2(x)\right),$$

where $f_1(x) = x^2$, $f_2(x) = (x-2)^2$, $x \in \left[-10^3, 10^3\right]$.

As stated in Table I *of* [13], *any point* $x \in [0,2]$ *is a Pareto optimal solution of this problem. Let* $X = \left[-10^3, 10^3\right]$. *Since each of the functions* $f_i, i = 1, 2$, *is continuously differentiable on* X, *which is closed and bounded, then each of* f_i *is locally Lipschitz continuous on* X. *Here* $\dfrac{\mathrm{d}f_1(x)}{\mathrm{d}x} = 2x$ *and*

$\dfrac{\mathrm{d}f_2(x)}{\mathrm{d}x} = 2(x-2)$. *Hence,* $\sup_{x \in X}\left|\dfrac{\mathrm{d}f_1(x)}{\mathrm{d}x}\right| \le 2000$ *and* $\sup_{x \in X}\left|\dfrac{\mathrm{d}f_2(x)}{\mathrm{d}x}\right| \le 2004$.

Therefore, we can take the Lipschitz constants $K_i = 2004$, $i = 1, 2$ *such that*

$$\left|f_i(y) - f_i(z)\right| \le K_i |y - z|, \text{for all } y, z \in X.$$

Let $\varepsilon = (\varepsilon_1, \varepsilon_2) = (50, 50)$. *Then, from* (27), *we have* $\eta = \dfrac{25}{1002}$. *In formula* (30), *let* $k_1 = 64 \times 10^3$. *Hence the cardinality of* Ω *is* $\mathrm{card}(\Omega) = 64 \times 10^3 + 1$ *and* $\dfrac{1}{k_1}(\beta_1 - \alpha_1) = \dfrac{1}{32}$, *and therefore inequality* (32) *is satisfied. Suppose that the population size is* $r = 200$. *For the stopping criterion, we take* $\delta = 0.99$ *and compute* $t^*_{\min}(\delta) = 5016$. *After* 5016 *iterations of Algorithm* 1, *the resulting set* $\overline{\Omega}$ *is the following*:

$$\overline{\Omega} = \left\{ \begin{array}{l} 0, 1, 2, \dfrac{1}{32}, \dfrac{1}{16}, \dfrac{3}{32}, \dfrac{1}{8}, \dfrac{5}{32}, \dfrac{3}{16}, \dfrac{7}{32}, \dfrac{1}{4}, \dfrac{9}{32}, \dfrac{5}{16}, \dfrac{11}{32}, \dfrac{3}{8}, \dfrac{13}{32}, \dfrac{7}{16}, \dfrac{15}{32}, \\[2mm] \dfrac{1}{2}, \dfrac{17}{32}, \dfrac{9}{16}, \dfrac{19}{32}, \dfrac{5}{8}, \dfrac{21}{32}, \dfrac{11}{16}, \dfrac{23}{32}, \dfrac{3}{4}, \dfrac{25}{32}, \dfrac{13}{16}, \dfrac{27}{32}, \dfrac{7}{8}, \dfrac{29}{32}, \dfrac{15}{16}, \\[2mm] \dfrac{31}{32}, \dfrac{33}{32}, \dfrac{17}{16}, \dfrac{35}{32}, \dfrac{9}{8}, \dfrac{37}{32}, \dfrac{19}{16}, \dfrac{39}{32}, \dfrac{5}{4}, \dfrac{41}{32}, \dfrac{21}{16}, \dfrac{43}{32}, \dfrac{11}{8}, \dfrac{45}{32}, \dfrac{23}{16}, \dfrac{47}{32}, \dfrac{7}{4}, \\[2mm] \dfrac{49}{32}, \dfrac{25}{16}, \dfrac{51}{32}, \dfrac{13}{8}, \dfrac{53}{32}, \dfrac{27}{16}, \dfrac{55}{32}, \dfrac{7}{4}, \dfrac{57}{32}, \dfrac{29}{16}, \dfrac{59}{32}, \dfrac{15}{8}, \dfrac{61}{32}, \dfrac{31}{16}, \dfrac{63}{32} \end{array} \right\}. \quad (34)$$

Remarks:

1) One should remember that the number $t^*_{\min}(\delta)$ depends on the prescribed probability δ. We have run Algorithm 1 many times up to 10,000 iterations and have observed the following changes in the set E_{t+1}: it has always become the set (34) somewhere between iterations 1155 and 1330, and has not changed in the later iterations. This means that the theoretically computed number of 5016 iterations gives the correct set $\overline{\Omega}$ (in the sense that it cannot be further improved), but in fact much less iterations are sufficient to obtain the same result.

2) The cardinality of $\bar{\Omega}$ is $\mathrm{card}(\bar{\Omega}) = 65$. Each element of $\bar{\Omega}$ belongs to the interval $[0, 2]$, and hence is a Pareto efficient solution.

3) According to the performance measure Diversity Metric Δ, see section B page 188 in [13], the mean and variance of Δ for Algorithm 1 is 0.1014490343 and 0.09539251009, respectively, where $d_f = d_l = 0$. Hence our algorithm finds better spread of solutions than any other algorithm included in Table III of [13], see **Figure 1**, this is because the mean is the smallest one.

Example 8 (Problem FON in Table I of [13]). *Consider the following optimization problem*:

$$\min\left(f_1(x), f_2(x)\right),$$

where

$$f_1(x) = 1 - \exp\left(-\sum_{i=1}^{3}\left(x_i - \frac{1}{\sqrt{3}}\right)^2\right), f_2(x) = 1 - \exp\left(-\sum_{i=1}^{3}\left(x_i + \frac{1}{\sqrt{3}}\right)^2\right),$$

with variable bounds $x_1, x_2, x_3 \in [-4, 4]$.

Table I *of* [13] *states that every point* (x_1, x_2, x_3) *satisfying the condition*
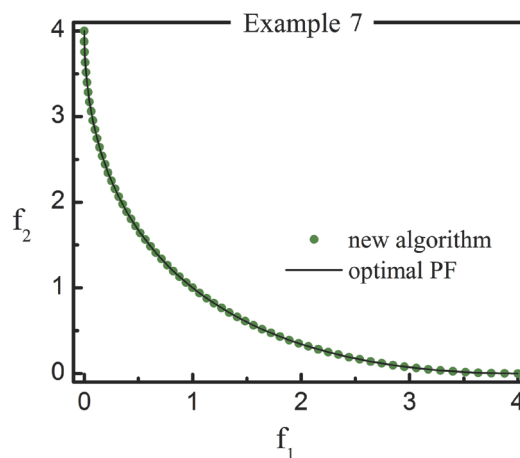
$$x_1 = x_2 = x_3 \in \left[-\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right] \tag{35}$$

is a Pareto optimal solution of this problem. Let $X = [-4, 4]^3$. *Since each of the functions* $f_i, i = 1, 2,$ *is continuously differentiable on* X, *which is closed and bounded, then each of* f_i *is locally Lipschitz continuous on* X. *We denote by* $\nabla f_i(x)$ *the gradient vector of* f_i *at* x:

$$\nabla f_i(x) = \left(\frac{\partial f_i(x)}{\partial x_1}, \frac{\partial f_i(x)}{\partial x_2}, \frac{\partial f_i(x)}{\partial x_3}\right)^{\mathrm{T}}, i = 1, 2.$$

Then

$$\left\|\nabla f_i(x)\right\|_\infty = \max_{1 \le j \le 3}\left|\frac{\partial f_i(x)}{\partial x_j}\right|, i = 1, 2.$$

Figure 1. True PF and nondominated solutions by New Algorithm on SCH.

Note that $\sup_{x\in X}\|\nabla f_i(x)\|_\infty \le 1$ *for* $i=1,2$. *For any* $y,z\in X$, *there exists* $u\in[y,z]$ *such that*

$$\left|f_i(y)-f_i(z)\right| = \left|\langle\nabla f_i(u),y-z\rangle\right| = \left|\sum_{j=1}^{3}\frac{\partial f_i(u)}{\partial x_j}(y_j-z_j)\right|$$

$$\le \sum_{j=1}^{3}\left|\frac{\partial f_i(u)}{\partial x_j}\right|\left|y_j-z_j\right| \le 3\|\nabla f_i(u)\|_\infty\|y-z\|_\infty \qquad (36)$$

$$\le 3\sup_{x\in X}\|\nabla f_i(x)\|_\infty\|y-z\|_\infty.$$

Therefore, we can take the Lipschitz constants $K_i=3$, $i=1,2$, *such that*

$$\left|f_i(y)-f_i(z)\right| \le K_i\|y-z\|_\infty, \text{ for all } y,z\in X. \qquad (37)$$

Let $\varepsilon=(\varepsilon_1,\varepsilon_2)=\left(\frac{3}{5},\frac{3}{5}\right)$. *Then, from* (27), *we have* $\eta=\frac{1}{5}$. *In formula* (30), *let* $k_i=50$, $i=1,2,3$. *Hence the cardinality of* Ω *is* $\mathrm{card}(\Omega)=51^3=132651$ *and* $\frac{1}{k_i}(\beta_i-\alpha_i)=\frac{4}{25}$, *and therefore inequality* (32) *is satisfied. Suppose that the population size is* $r=200$. *For the stopping criterion, we take* $\delta=0.99$ *and compute* $t^*_{\min}(\delta)=10878$. *After* 10878 *iterations of Algorithm* 1, *the resulting set* $\overline{\Omega}$ *is the following:*

$$\overline{\Omega}=\left\{\begin{array}{l}
(0,0,0),\left(\frac{4}{25},\frac{4}{25},\frac{4}{25}\right),\left(-\frac{4}{25},-\frac{4}{25},-\frac{4}{25}\right),\left(\frac{8}{25},\frac{8}{25},\frac{8}{25}\right),\left(-\frac{8}{25},-\frac{8}{25},-\frac{8}{25}\right), \\[2mm]
\left(\frac{12}{25},\frac{12}{25},\frac{12}{25}\right),\left(-\frac{12}{25},-\frac{12}{25},-\frac{12}{25}\right),\left(\frac{16}{25},\frac{16}{25},\frac{16}{25}\right),\left(-\frac{16}{25},-\frac{16}{25},-\frac{16}{25}\right),\left(\frac{4}{25},\frac{4}{25},\frac{8}{25}\right), \\[2mm]
\left(\frac{8}{25},\frac{4}{25},\frac{4}{25}\right),\left(\frac{4}{25},\frac{8}{25},\frac{4}{25}\right),\left(-\frac{4}{25},-\frac{4}{25},-\frac{8}{25}\right),\left(-\frac{8}{25},-\frac{4}{25},-\frac{4}{25}\right),\left(-\frac{4}{25},-\frac{8}{25},-\frac{4}{25}\right), \\[2mm]
\left(\frac{4}{25},\frac{8}{25},\frac{8}{25}\right),\left(\frac{8}{25},\frac{8}{25},\frac{4}{25}\right),\left(\frac{8}{25},\frac{4}{25},\frac{8}{25}\right),\left(-\frac{4}{25},-\frac{8}{25},-\frac{8}{25}\right),\left(-\frac{8}{25},-\frac{8}{25},-\frac{4}{25}\right), \\[2mm]
\left(-\frac{8}{25},-\frac{4}{25},-\frac{8}{25}\right),\left(\frac{12}{25},\frac{8}{25},\frac{8}{25}\right),\left(\frac{8}{25},\frac{8}{25},\frac{12}{25}\right),\left(\frac{8}{25},\frac{12}{25},\frac{8}{25}\right),\left(-\frac{12}{25},-\frac{8}{25},-\frac{8}{25}\right), \\[2mm]
\left(-\frac{8}{25},-\frac{8}{25},-\frac{12}{25}\right),\left(-\frac{8}{25},-\frac{12}{25},-\frac{8}{25}\right),\left(\frac{12}{25},\frac{12}{25},\frac{8}{25}\right),\left(\frac{8}{25},\frac{12}{25},\frac{12}{25}\right),\left(\frac{12}{25},\frac{8}{25},\frac{12}{25}\right), \\[2mm]
\left(-\frac{12}{25},-\frac{12}{25},-\frac{8}{25}\right),\left(-\frac{8}{25},-\frac{12}{25},-\frac{12}{25}\right),\left(-\frac{12}{25},-\frac{8}{25},-\frac{12}{25}\right),\left(\frac{12}{25},\frac{12}{25},\frac{16}{25}\right),\left(\frac{16}{25},\frac{12}{25},\frac{12}{25}\right), \\[2mm]
\left(\frac{12}{25},\frac{16}{25},\frac{12}{25}\right),\left(-\frac{12}{25},-\frac{12}{25},-\frac{16}{25}\right),\left(-\frac{16}{25},-\frac{12}{25},-\frac{12}{25}\right),\left(-\frac{12}{25},-\frac{16}{25},-\frac{12}{25}\right),\left(\frac{12}{25},\frac{16}{25},\frac{16}{25}\right), \\[2mm]
\left(\frac{16}{25},\frac{16}{25},\frac{12}{25}\right),\left(\frac{16}{25},\frac{12}{25},\frac{16}{25}\right),\left(-\frac{12}{25},-\frac{16}{25},-\frac{16}{25}\right),\left(-\frac{16}{25},-\frac{16}{25},-\frac{12}{25}\right),\left(-\frac{16}{25},-\frac{12}{25},-\frac{16}{25}\right), \\[2mm]
\left(\frac{4}{25},\frac{4}{25},0\right),\left(0,\frac{4}{25},\frac{4}{25}\right),\left(\frac{4}{25},0,\frac{4}{25}\right),\left(-\frac{4}{25},-\frac{4}{25},0\right),\left(0,-\frac{4}{25},-\frac{4}{25}\right),\left(-\frac{4}{25},0,-\frac{4}{25}\right), \\[2mm]
\left(\frac{4}{25},0,0\right),\left(0,0,\frac{4}{25}\right),\left(0,\frac{4}{25},0\right),\left(-\frac{4}{25},0,0\right),\left(0,0,-\frac{4}{25}\right),\left(0,-\frac{4}{25},0\right)
\end{array}\right\}. \qquad (38)$$

Remarks:

1) In practical tests, the set E_{t+1} has always become the set (38) somewhere between iterations 3475 and 3500, and has not changed in the later iterations.

2) The cardinality of $\overline{\Omega}$ is $\mathrm{card}(\overline{\Omega})=57$. The points in $\overline{\Omega}$ which satisfy condition (35) are Pareto optimal but the other elements of $\overline{\Omega}$ are not optimal. However, it follows from Proposition 6 that all elements of $\overline{\Omega}$ are ε-efficient

solutions with probability δ.

3) According to the performance measure Diversity Metric Δ, see section B page 188 in [13], the mean and variance of Δ for Algorithm 1 is 0.06078996663 and 0.4859115201, respectively, where $d_f = d_l = 0.01343253265$. Hence our algorithm finds better spread of solutions than any other algorithm included in Table III of [13], see **Figure 2**, this is because the mean is the smallest one.

Example 9 (Problem POL in Table I of [13]).

$$\min\left(f_1(x), f_2(x)\right),$$

where

$$f_1(x) = \left[1 + (A_1 - B_1)^2 + (A_2 - B_2)^2\right], \quad f_2(x) = \left[(x_1 + 3)^2 + (x_2 + 1)^2\right],$$

$$A_1 = 0.5\sin 1 - 2\cos 1 + \sin 2 - 1.5\cos 2, \quad A_2 = 1.5\sin 1 - \cos 1 + 2\sin 2 - 0.5\cos 2,$$

$$B_1 = 0.5\sin x_1 - 2\cos x_1 + \sin x_2 - 1.5\cos x_2,$$

$B_2 = 1.5\sin x_1 - \cos x_1 + 2\sin x_2 - 0.5\cos x_2$, *with variable bounds* $x_1, x_2 \in [-\pi, \pi]$.

POL is a problem with two nonconvex Pareto fronts that are disconnected in both the objective and decision variable spaces, see [13]. *The true set of Pareto-optimal solutions is difficult to know in this problem.* **Figure 3** *illustrates that Algorithm* 1 *is able to discover the two disconnected Pareto fronts that lie on the boundaries of the search space.*

Let $X = [-\pi, \pi] \times [-\pi, \pi]$. *Since each of the functions* $f_i, i = 1, 2$, *is continuously differentiable on* X, *which is closed and bounded, then each of* $f_i, i = 1, 2$, *is locally Lipschitz continuous on* X. *By using a computer program, it is possible to show that* $\sup_{x \in X} \left\|\nabla f_1(x)\right\|_\infty \leq 34$ *and* $\sup_{x \in X} \left\|\nabla f_2(x)\right\|_\infty \leq 13$. *Using an estimate similar to* (36), *but with two variables, we find that, for the following Lipschitz constants:* $K_1 = 68$, *and* $K_2 = 26$, *we have*

$$\left|f_i(y) - f_i(z)\right| \leq K_i \left\|y - z\right\|_\infty, \text{ for all } y, z \in X, i = 1, 2.$$
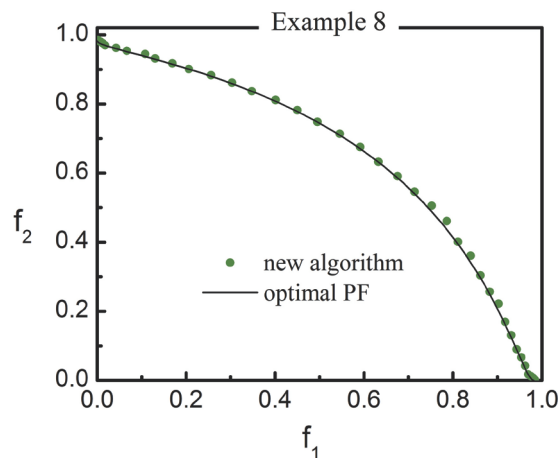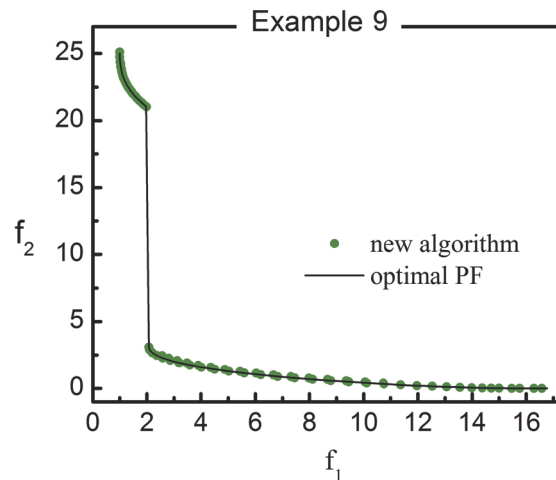

Figure 2. True PF and nondominated solutions by New Algorithm on FON.

Let $\varepsilon = (\varepsilon_1, \varepsilon_2) = (5/2, 1)$. *Then*, from (27), *we have* $\eta = \dfrac{5}{136}$. *In formula*

(30), *let* $k_i = 100$, $i = 1, 2$. *Hence the cardinality of* Ω *is*

$\text{card}(\Omega) = 101^2 = 10201$ *and* $\dfrac{1}{k_i}(\beta_i - \alpha_i) = \dfrac{\pi}{50}$, *and therefore inequality (32) is*

satisfied. Suppose that the population size is $r = 200$. *For the stopping criterion,*
we take $\delta = 0.99$ *and compute* $t^*_{\min}(\delta) = 706$. *After* 706 *iterations of Algorithm*
1, *the resulting set* $\bar{\Omega}$ *is the following:*

$$\bar{\Omega} = \left\{ \begin{array}{c} (-\pi, 0), \left(-\pi, \frac{1}{50}\pi\right), \left(-\pi, \frac{1}{25}\pi\right), \left(-\pi, \frac{3}{50}\pi\right), \left(-\pi, \frac{2}{25}\pi\right), \\[4pt] \left(-\pi, \frac{1}{10}\pi\right), \left(-\pi, \frac{3}{25}\pi\right), \left(-\pi, \frac{7}{50}\pi\right), \left(-\pi, \frac{4}{25}\pi\right), \\[4pt] \left(-\pi, \frac{9}{50}\pi\right), \left(-\pi, \frac{1}{5}\pi\right), \left(-\pi, \frac{11}{50}\pi\right), \left(-\pi, \frac{6}{25}\pi\right), \\[4pt] \left(-\pi, -\frac{1}{50}\pi\right), \left(-\pi, -\frac{1}{25}\pi\right), \left(-\pi, -\frac{3}{50}\pi\right), \left(-\pi, -\frac{2}{25}\pi\right), \\[4pt] \left(-\pi, -\frac{1}{10}\pi\right), \left(-\pi, -\frac{3}{25}\pi\right), \left(-\pi, -\frac{7}{50}\pi\right), \left(-\pi, -\frac{4}{25}\pi\right), \\[4pt] \left(-\pi, -\frac{9}{50}\pi\right), \left(-\pi, -\frac{1}{5}\pi\right), \left(-\pi, -\frac{11}{50}\pi\right), \\[4pt] \left(-\pi, -\frac{6}{25}\pi\right), \left(-\pi, -\frac{13}{50}\pi\right), \left(-\pi, -\frac{7}{25}\pi\right), \left(-\pi, -\frac{3}{10}\pi\right), \\[4pt] \left(-\pi, -\frac{8}{25}\pi\right), \left(\frac{13}{50}\pi, \frac{1}{2}\pi\right), \left(\frac{13}{50}\pi, \frac{13}{25}\pi\right), \left(\frac{13}{50}\pi, \frac{27}{50}\pi\right), \\[4pt] \left(\frac{17}{50}\pi, \frac{13}{25}\pi\right), \left(\frac{17}{50}\pi, \frac{27}{50}\pi\right), \left(\frac{8}{25}\pi, \frac{1}{2}\pi\right), \left(\frac{8}{25}\pi, \frac{13}{50}\pi\right), \\[4pt] \left(\frac{8}{25}\pi, \frac{27}{50}\pi\right), \left(\frac{8}{25}\pi, \frac{14}{25}\pi\right), \left(\frac{8}{25}\pi, \frac{29}{50}\pi\right), \left(\frac{8}{25}v, \frac{3}{5}\pi\right), \\[4pt] \left(\frac{8}{25}\pi, \frac{31}{50}\pi\right), \left(\frac{8}{25}\pi, \frac{16}{25}\pi\right), \left(\frac{7}{25}\pi, \frac{12}{25}\pi\right), \left(\frac{7}{25}\pi, \frac{1}{2}\pi\right), \\[4pt] \left(\frac{7}{25}\pi, \frac{13}{25}\pi\right), \left(\frac{7}{25}\pi, \frac{27}{50}\pi\right), \left(\frac{6}{25}\pi, \frac{13}{25}\pi\right), \left(-\frac{49}{50}\pi, 0\right), \\[4pt] \left(-\frac{49}{50}\pi, \frac{1}{50}\pi\right), \left(-\frac{49}{50}\pi, \frac{1}{25}\pi\right), \left(-\frac{49}{50}\pi, \frac{3}{50}\pi\right), \left(-\frac{49}{50}\pi, \frac{2}{25}\pi\right), \\[4pt] \left(-\frac{49}{50}\pi, \frac{1}{10}\pi\right), \left(-\frac{49}{50}\pi, \frac{3}{25}\pi\right), \left(-\frac{49}{50}\pi, \frac{7}{50}\pi\right), \\[4pt] \left(-\frac{49}{50}\pi, \frac{4}{25}\pi\right), \left(-\frac{49}{50}\pi, \frac{9}{50}\pi\right), \left(-\frac{49}{50}\pi, -\frac{1}{50}\pi\right), \left(-\frac{49}{50}\pi, -\frac{1}{25}\pi\right), \\[4pt] \left(-\frac{49}{50}\pi, -\frac{3}{50}\pi\right), \left(-\frac{49}{50}\pi, -\frac{2}{25}\pi\right), \left(-\frac{49}{50}\pi, -\frac{1}{10}\pi\right), \\[4pt] \left(-\frac{49}{50}\pi, -\frac{3}{25}\pi\right), \left(-\frac{49}{50}\pi, -\frac{3}{10}\pi\right), \left(-\frac{49}{50}\pi, -\frac{8}{25}\pi\right), \left(\frac{3}{10}\pi, \frac{12}{25}\pi\right), \\[4pt] \left(\frac{3}{10}\pi, \frac{1}{2}\pi\right), \left(\frac{3}{10}\pi, \frac{13}{25}\pi\right), \left(\frac{3}{10}\pi, \frac{27}{50}\pi\right), \\[4pt] \left(\frac{3}{10}\pi, \frac{14}{25}\pi\right), \left(\frac{3}{10}\pi, \frac{29}{50}\pi\right), \left(\frac{3}{10}\pi, \frac{3}{25}\pi\right), \left(\frac{3}{10}\pi, \frac{31}{50}\pi\right), \\[4pt] \left(-\frac{24}{10}\pi, -\frac{8}{25}\pi\right), \left(-\frac{24}{10}\pi, -\frac{3}{10}\pi\right) \end{array} \right\}. \qquad (39)$$

Remarks:

1) In practical tests, the set E_{t+1} has always become the set (39) somewhere

Figure 3. True PF and nondominated solutions by New Algorithm on POL.

between iterations 285 and 350, and has not changed in the later iterations.

2) The cardinality of $\overline{\Omega}$ is $\mathrm{card}(\overline{\Omega}) = 75$. It follows from Proposition 6 that all elements of $\overline{\Omega}$ are ε-efficient solutions with probability δ.

3) According to the performance measure Diversity Metric Δ, see section B page 188 in [13], the mean and variance of Δ for Algorithm 1 is 0.5021982345 and 0.7382353788, respectively, where $d_f = 0.1063348336$ and $d_l = 0.01974325126$ for the left Pareto front in **Figure 3**, and $d_f = 0.0139738762$ and $d_l = 0.1941428847$ for the right Pareto front in **Figure 3**. Hence, in Table III of [13], a better spread of solution is achieved by the algorithm NSGA-II (real-coded) in [13]. The spread of solution by our algorithm is the next-best for this problem.

6. Conclusion

We have presented a new evolutionary method for generating ε-efficient solutions of a continuous multiobjective programming problem. This was achieved by discretizing the problem and then using a genetic algorithm. Some probabilistic stopping criteria were used for this procedure to obtain, with a prescribed probability, all minimal solutions for the discretized problem, which are ε-efficient solutions for the original problem. This article contains the main underlying theory and only some preliminary numerical computations pertaining to this method.

Acknowledgements

References

[1] Ruzika, S. and Wiecek, M.M. (2005) Approximation Methods in Multiobjective Programming. *Journal of Optimization Theory and Applications*, **126**, 473-501.

https://doi.org/10.1007/s10957-005-5494-4

[2] Ghaznavi-Ghosoni, B.A., Khorram, E. and Soleimani-damaneh, M. (2013) Scalarization for Characterization of Approximate Strong/Weak/Proper Efficiency in Multi-Objective Optimization. *Optimization*, **62**, 703-720.
https://doi.org/10.1080/02331934.2012.668190

[3] Loridan, P. (1984) ε-Solutions in Vector Minimization Problems. *Journal of Optimization Theory and Applications*, **42**, 265-276.
https://doi.org/10.1007/BF00936165

[4] Engau, A. and Wiecek, M.M. (2007) Generating ε-Efficient Solutions in Multiobjective Programming. *European Journal of Operational Research*, **177**, 1566-1579.
https://doi.org/10.1016/j.ejor.2005.10.023

[5] Engau, A. and Wiecek, M.M. (2007) Exact Generation of Epsilon-Efficient Solutions in Multiple Objective Programming. *OR Spectrum*, **29**, 335-350.
https://doi.org/10.1007/s00291-006-0044-5

[6] Laumanns, M., Thiele, L., Deb, K. and Zitzler, E. (2002) Combining Convergence and Diversity in Evolutionary Multiobjective Optimization. *Evolutionary Computation*, **10**, 263-282. https://doi.org/10.1162/106365602760234108

[7] Schutze, O., Laumanns, M., Tantar, E., Coello Coello, C.A. and Talbi, E.-G. (2007) Convergence of Stochastic Search Algorithms to Gap-Free Pareto Front Approximations. *Proceedings of the Genetic and Evolutionary Computation Conference* (*GECCO*-2007), 892-899.

[8] Studniarski, M. (2011) Finding All Minimal Elements of a Finite Partially Ordered Set by Genetic Algorithm with a Prescribed Probability. *Numerical Algebra*, *Control and Optimization*, **1**, 389-398. https://doi.org/10.3934/naco.2011.1.389

[9] Vose, M.D. (1999) The Simple Genetic Algorithm: Foundations and Theory. MIT Press, Cambridge.

[10] Reeves, C.R. and Rowe, J.E. (2003) Genetic Algorithms—Principles and Perspectives: A Guide to GA Theory. Kluwer, Boston.

[11] Rudolph, G. and Agapie, A. (2000) Convergence Properties of Some Multi-Objective Evolutionary Algorithms. In: Zalzala, A., *et al.*, Eds., *Proceedings of the* 2000 *Congress on Evolutionary Computation* (*CEC* 2000), Vol. 2, IEEE Press, Piscataway, 1010-1016.

[12] Osman, M.S., Abo-Sinna, M.A. and Mousa, A.A. (2005) An Effective Genetic Algorithm Approach to Multiobjective Resource Allocation Problems (MORAPs). *Applied Mathematics and Computation*, **163**, 755-768.
https://doi.org/10.1016/j.amc.2003.10.057

[13] Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T. (2002) A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, **6**, 182-197. https://doi.org/10.1109/4235.996017