Int. J. Appl. Math. Comput. Sci., 2006, Vol. 16, No. 1, 59–84

NICHING MECHANISMS IN EVOLUTIONARY COMPUTATIONS

ZDZISŁAW KOWALCZUK, TOMASZ BIAŁASZEWSKI

Faculty of Electronics, Telecommunications and Computer Science
Gdańsk University of Technology
ul. Narutowicza 11/12, 80–952 Gdańsk, Poland
e-mail: kova@pg.gda.pl

Different types of niching can be used in genetic algorithms (GAs) or evolutionary computations (ECs) to sustain the diversity of the sought optimal solutions and to increase the effectiveness of evolutionary multi-objective optimization solvers. In this paper four schemes of niching are proposed, which are also considered in two versions with respect to the method of invoking: a continuous realization and a periodic one. The characteristics of these mechanisms are discussed, while as their performance and effectiveness are analyzed by considering exemplary multi-objective optimization tasks both of a synthetic and an engineering (FDI) design nature.

Keywords: niching, ranking, Pareto-optimality, genetic algorithms, evolutionary computations, multi-objective optimization, solutions diversity, engineering designs, detection observers

1. Introduction

Natural selection plays a critical role in optimizing mechanisms of rivalry between individuals and species. It determines if a given individual or species or whole populations will survive. The selected, adapted individuals of a high fitness have greater chances of producing offspring of desirable features at least similar to parental ones. As a result of such a continual selection process, new generations of improved suitability are created. To our surprise, nature sometimes admits (with small probability) the survival of individuals weakly fitted. Such individuals establish the source of strewn information helping the diversity of the current population, which allows inserting innovations into the genetic information transmitted to the next generations. In other words, natural selection in certain situation tests also other, seemingly non-optimal, possibilities of achieving befitting individuals. Nevertheless, in nature the time for protecting weak individuals is usually very short, and, as a result, they promptly disappear.

On a similar basis, in a synthetic environment one can impose a policy of breeding weak species in order to sustain them. In genetic algorithms (Chambers, 1995; Forrest, 1993; Goldberg, 1986; 1989; Grefenstette, 1985; Holland, 1975; Kirstinsson, 1992; Man *et al.*, 1997; Michalewicz, 1996) and their engineering applications (Cotta and Schaefer, 2004; Fogarty and Bull, 1995; Huang and Wang, 1997; Li *et al.*, 1997; Korbicz *et al.*, 2004; Linkens and Nyongensa, 1995; Martinez *et al.*, 1996; Obuchowicz and Prętki, 2004; Park and Kandel, 1994), niching can be recognized as a mechanism that preserves (apart from the best individuals in terms of fitness) also average and worse individuals in order to sustain diverse generations (Goldberg, 1989; 1990; Kowalczuk *et al.*, 1999a; Kowalczuk and Białaszewski, 2004a; Michalewicz, 1996; Ryan, 1995). By inserting these individuals into a parental pool they are given a chance to relay their diverse genetic codes to their offspring. The mechanism balances the populations of the existing species by increasing the chance of mating for the individuals from sparse (weaker) niches and decreasing that chance for the ones from dense niches (species). In effect, such a mechanism of 'uniform breeding' prevents genetic algorithms (GA) or evolutionary computations (EC) from premature convergence, as well as supports their ability to adapt.

Evolutionary computations are most appreciated in solving difficult multi-objective optimization problems. In such cases the concept of optimality in the Pareto sense is usually applied when assessing the merit of solutions (Goldberg, 1989; 1990; Kowalczuk *et al.*, 1999a; 1999b; Kowalczuk and Białaszewski, 2004a; Man *et al.*, 1997; Michalewicz, 1996). With maximization tasks in mind, the solutions can be classified as dominated and non-dominated (Pareto optimal). Next, according to such an assessment, in a ranking process each individual is assigned a scalar rank representing its degree of domination.

In this paper we reflect on different techniques for niching, which we segregate with respect to the object of the niching 'manipulation' (Kowalczuk and Białaszewski, 2000b; 2004a; 2004b). In such a way four techniques can be distinguished: the niching of the fitness (NF) and the ranks (NR) of all the individuals in a generation, and the niching of the fitness (NFP) and the ranks (NRP) of the parents solely. Furthermore, these four types of niching are considered in two versions with respect to the frequency of their application within a GA/EC cycle. The efficacy of these mechanisms is consequently verified in solving exemplary multi-objective optimal design problems, including linear state observers for detection systems.

2. Multi–Objective Optimization

As is observed in nature, also in engineering problems we frequently specify a set of criteria with respect to which a given product is optimized. In practical terms the issue often boils down to making a choice from amongst several equivalent 'optimal' solutions or to trading off between different criteria of performance, like price, reliability, safety, etc. The issue of total optimization of several objective functions at the same time is therefore most essential (Dridi and Kacem, 2004; Kowalczuk *et al.*, 1999a; Kowalczuk and Białaszewski, 2004a). Such types of designing problems are often called multi-objective optimization (MO) tasks (Goldberg, 1989; Michalewicz, 1996; Viennet *et al.*, 1996).

2.1. Multi–Objective Optimization Problem

Consider the following n-dimensional vector $\boldsymbol{x}$ of the parameters searched for:

$$\boldsymbol{x} = \begin{bmatrix} x_1 & x_2 & \ldots & x_n \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^n, \quad n \in \mathbb{N}, \tag{1}$$

which is assessed according to an m-dimensional vector $\boldsymbol{f}(\boldsymbol{x})$ of objective functions $f_j(\boldsymbol{x})$, $j = 1, 2, \ldots, m$:

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix} f_1(\boldsymbol{x}) & f_2(\boldsymbol{x}) & \ldots & f_m(\boldsymbol{x}) \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^m, \tag{2}$$

$m \in \mathbb{N}$. Assuming, for simplicity, that all coordinates of the criterion vector (2) represent profit functions, the multi-objective optimization task studied can be formulated as a multi-profit maximization task without constraints:

$$\max_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}). \tag{3}$$

2.2. Multi–Objective Optimization Methods

A method of solving multi-objective problems can be found on the basis of a simple integrated criterion or by taking into consideration another interconnected assessment. Such methods of solving MO optimality problems can thus be divided into two groups, representing a classical approach (Chen *et al.*, 1996; Goldberg, 1989; Michalewicz, 1996; Zakian and Al-Naib, 1973) and a scheduling or ranking one (Goldberg, 1989; Kowalczuk *et al.*, 1999a; Kowalczuk and Białaszewski, 2004a; Man *et al.*, 1997; Michalewicz, 1996).

Within the classical multi-objective optimization ways we place the methods considering:

– weighted profits (Michalewicz, 1996),

– distance functions (Michalewicz, 1996), and

– sequential inequalities (Chen *et al.*, 1996; Zakian and Al-Naib, 1973),

whereas the ranking approaches are based on:

– Pareto optimality ranking (Goldberg, 1989; Michalewicz, 1996; Man *et al.*, 1997), or

– other more distinctive and utilitarian indices, like the global optimality level proposed in (Kowalczuk and Białaszewski, 2000a; 2004a).

They can be characterized as follows:

- In the method of weighted profits all coordinates of the profit vector $\boldsymbol{f}(\boldsymbol{x})$ are combined into one profit function by means of a normalized vector of weights.

- The distance function method consists in integrating the coordinates of the distance from the profit function vector $\boldsymbol{f}(\boldsymbol{x})$ to a certain demand vector into a scalar profit function $h(\boldsymbol{x})$ measured by means of a simple (Euclidean) vector norm.

- The method of sequential inequalities (Zakian and Al-Naib, 1973) uses a transformation of the analyzed multi-profit maximization task (without constraints) into a set of tasks of maximization of the partial profit functions with inequality constraints suitably parameterized; which means searching for a parameter vector $\boldsymbol{x}$, which does not violate a fixed set of inequalities (Chen *et al.*, 1996; Kowalczuk and Białaszewski, 2004a).

The above classical methods are simple. They have, however, the disadvantage of relying on an arbitrary choice of a parameter (the weighting vector, demand vector or limit values for the profit functions). In effect, the obtained solutions are clearly conditioned by the weights, demand or limits applied, which means an obvious simplification of the multi-profit maximization problem. What is more, for the purpose of the integration of separate objectives into one measure, the designer has to decide on the relations (or significance) among the objectives, which can be a cumbersome task.

In the last two decades a wealth of various evolutionary multi-objective optimization (EMO) methods (Coello, 2001; Forrest, 1993; Goldberg, 1989; Grefenstette, 1985; Hajela and Lin, 1992; Horn *et al.*, 1994; Korbicz *et al.*, 2004; Schaffer, 1985; Srinivas and Deb, 1994; Viennet *et al.*, 1996; Zitzler and Thiele, 1998) were proposed for solving multi-objective problems in multi-dimensional spaces more effectively. Among them, one can enumerate the following GA/EA algorithms:

– Vector Evaluated GAs (VEGAs),

– Hajela and Lin GAs (HLGAs),

– Multi-Objective GAs (MOGAs),

– Niched Pareto GAs (NPGAs),

– Non-dominated Sorting GAs (NSGAs, NSGA2s),

– Strength Pareto EAs (SPEAs and SPEA2s),

– Genetic Gender Algorithms (GGAs).

The **VEGA** (Schaffer, 1985) differs from the basic GA in the process of selection. A separate parental pool is there chosen by taking into account each single objective function and by making an independent selection proportional to this single fitness measure. The aggregate parental pool (with repeated individuals) constitutes a genetic material for creating new individuals by means of the operations of crossover and mutation. This is a non-Pareto approach, which can be easily implemented. Though, generally, its optimal results have a disadvantage of being superior only with respect to one objective of the multi-objective task considered.

The **HLGA** (Hajela and Lin, 1992), representing another non-Pareto approach, uses the method of weighted profits, where all co-ordinates of the profit vector $\boldsymbol{f}(\boldsymbol{x})$ are combined into one scalar function

$$g(\boldsymbol{x}) = \boldsymbol{w} \cdot \boldsymbol{f}(\boldsymbol{x}),$$

with $\boldsymbol{w} = [\, w_1 \quad w_2 \quad \dots \quad w_m \,] \in \mathbb{R}^m$ denoting a vector of normalized weights $w_i \in [0,\, 1]$, $\sum_{i=1}^{m} w_i = 1$. This time though, the weights encoded in the sought genotype are also optimized:

$$\max_{\boldsymbol{x},\boldsymbol{w}} g(\boldsymbol{x}, \boldsymbol{w}). \tag{4}$$

The HLGA can be easily combined with some mating restrictions and a niching mechanism, utilized for sustaining certain diversity among the solutions.

Within the **MOGA** approach (Fonseca and Fleming, 1993; Forrest, 1993), each solution is assigned a certain scalar quantity called ranking, which is proportional to the number of individuals in the current population over which the analyzed individual dominates in the sense of Pareto. The population is sorted according to this ranking, and each individual is then assigned an auxiliary fitness value by suitably interpolating the ranking from the best to the worst solutions. Niching can be additionally incorporated in order to avoid premature convergence.

The **NPGA** (Horn and Nafpliotis, 1993; Horn *et al.*, 1994) uses a tournament selection based on the Pareto optimality, where in the process of creating a parental pool, two randomly chosen individuals conquer with the members of a third party, being some stochastically chosen set of solutions. The winner of this comparative tournament is this individual which is non-dominated with respect to all members of this set. When both competing individuals are equal (dominated or not dominated), the result of the tournament depends on the number of individuals in their niche defined in the objective space. Explicitly, the individual from a sparser niche is selected for the constructed parental pool.

The **NSGA** proposed by (Srinivas and Deb, 1994) also relays on the concept of the Pareto domination (ranking) performed on the analyzed population of individuals. All non-dominated individuals are assigned an artificial (dummy) fitness measure, which is proportional to the population size. Next, the selected non-dominated individuals are removed from the analyzed population. The remaining individuals in the population are classified similarly (in the Pareto sense). The process continues until all individuals are evaluated. Again, to sustain the diversity within the classified individuals (making a current Pareto front) their artificial fitness values are subject to warping (the so-called niching). A general (full-cycle) conception of the NSGA approach is presented as Procedure 2.1.

Procedure 2.1. Non-dominated Sorting GA
Generate randomly an initial population V of N individuals $\{\boldsymbol{x}_i\}_{i=1}^{N}$
while $t \leq t\max$
  Pareto _ front $\leftarrow 1$
  while not all individuals are classified (in the Pareto sense)
    Determine the non-dominated individuals (dominators)
    Assign the artificial fitness: $\boldsymbol{x}_i \rightarrow \boldsymbol{f}(\boldsymbol{x}_i)$
    Warp (niche) the fitness of individuals: $\boldsymbol{f}(\boldsymbol{x}_i) \rightarrow \tilde{f}(\boldsymbol{x}_i)$
    Estimate the rank of individuals: $\tilde{f}(\boldsymbol{x}_i) \rightarrow \tilde{r}(\boldsymbol{x}_i)$
    Pareto _ front $\leftarrow$ Pareto _ front+1
  end while
  Select the parents {proportionate/stochastic} according to $\tilde{r}(\mathbf{x}_i)$
  Create the new offspring V' by making:
    Crossover
    Mutation

Replace the population: $V \rightarrow V'$
Cycle: $t \leftarrow t+1$
end (of NSGA)

The improved algorithm **NSGA-2** uses the elitism strategy and the crowding mechanism (De Jong, 1975), which keeps the diversity of the population without specifying additional parameters (Deb *et al.*, 2000). Within the elitism strategy one considers an aggregate population containing the offspring and parent generations (N offspring and N parents), combines them, and classifies them with respect to their Pareto optimality. Since such a population has size $2N$, only the first N, most optimal individuals are chosen in a simple selection process based on a scalar measure of the crowding distance (which is productively applied only with respect to the members of the lowest Pareto front accepted). Specifically, the crowding distance is defined as an aggregate size of a minimum hypercube containing two neighboring individuals in the objective space. Hence the individuals of an analyzed front can be sorted by considering the crowding distance in descending order. Consequently, in this case it is the crowding distance mechanism which is responsible for sustaining diversity. The crowding distance assignment for the members of each Pareto front, used within the NSGA2 algorithm, is presented in the form of Procedure 2.2 below.

Procedure 2.2. Crowding Distance Assignment
Choose the individuals of the same (highest) rank $r(\boldsymbol{x}_i)$
Set the crowding distance of each individual $\boldsymbol{x}_i$ to zero: $d(\boldsymbol{x}_i) \leftarrow 0$
for (each partial profit) $k=1$ **to** m
  Sort the non-dominated individuals with respect to f_k: $\min\{f_k\} \longrightarrow \max\{f_k\}$
  Fix the vector s^k of the indices of the sorted individuals
  Set $d(\boldsymbol{x}_{s_1^k}) \leftarrow \infty$ and $d(\boldsymbol{x}_{s_l^k}) \leftarrow \infty$
    (l is the number of dominators)
    for $j=2$ **to** $(l-1)$
      $d(\boldsymbol{x}_{s_j^k}) \leftarrow d(\boldsymbol{x}_{s_j^k}) + \left[f_k(\boldsymbol{x}_{s_{j+1}^k}) - f_k(\boldsymbol{x}_{s_{j-1}^k})\right]$
    end for
end for
end (of crowding estimation)

As a result of the Pareto ranking and crowding distance assignments, each individual has two more attributes: the rank $r(\boldsymbol{x}_i)$ and the crowding distance $d(\boldsymbol{x}_i)$, which are suitably used in the NSGA2 algorithm overviewed in the form of Procedure 2.3.

Procedure 2.3. Non-dominated Sorting GA 2
Generate randomly an initial population V of N individuals $\{\mathbf{x}_i\}_{i=1}^N$
Pareto _front $\leftarrow 1$
while not all (N) individuals are yet classified in Pareto sense
  Determine the non-dominated individuals (dominators)
  Assign the rank (auxiliary fitness): $\boldsymbol{x}_i \rightarrow r(\boldsymbol{x}_i)$
  Delete the analyzed Pareto front
  Pareto _front $\leftarrow$ Pareto _front+1
end while (the initial rank assignment)
while $t \leq t\max$
  Select the parental pool (by the tournament principle)
  Create the (new) offspring V' by making:
    Crossover
    Mutation
  Aggregate V and V' together
  Pareto _ front $\leftarrow 1$
    while not all $(2N)$ individuals are yet Pareto-classified
      Determine the non-dominated solutions
      Assign the auxiliary fitness: $\boldsymbol{x}_i \rightarrow r(\boldsymbol{x}_i)$
      Compute the crowding distance: $\mathbf{x}_i \rightarrow d(\boldsymbol{x}_i)$
      Sort the individuals of the Pareto front w.r.t. $d(\boldsymbol{x}_i)$
      Delete the analyzed Pareto front
      Pareto _front $\leftarrow$ Pareto _front+1
    end while (sorting)
  Choose the first N individuals (of highest rank and distance) to the next generation V''
  Replace the population: $V \leftarrow V''$
  Cycle: $t \leftarrow t+1$
end (of NSGA2)

In the **SPEA**, Zitzler and Thiele (1998) consider two sets of solutions: a regular population and an external (archive) one. The algorithm begins with an initial population and an empty external set. In each evolutionary cycle, all non-dominated individuals are copied into the external population. During an update operation, all dominated and duplicated individuals are removed from the archive. When the number of non-dominated solutions in the current generation exceeds a fixed archival limit, certain non-dominated individuals are deleted from the archive by using a truncation (clustering) method, which preserves the principal characteristics of the set. Each individual in the external population is assigned a strength value. The strength value $\sigma(\boldsymbol{x}_j) \in [0,1)$ is a normalized number of the population members that, in the Pareto sense, are dominated by (or equal to) the solution $\boldsymbol{x}_j$ being considered. A weakness (inverse fitness) function $r(\boldsymbol{x}_i)$ of an individual $(\boldsymbol{x}_i)$ of the regular population is computed by summing the strengths $\sigma(\boldsymbol{x}_j)$ of all the archive members that dominate it (or are equal to $\boldsymbol{x}_i$). The parental pool is composed by using the binary tournament selection, taking into account the individuals from the aggregate population (the regular and external ones set together).

To overcome potential drawbacks of the SPEA, in a **SPEA2** modification (Zitzler *et al.*, 2001) the strength value $\sigma(\boldsymbol{x}_i)$ is assigned to all the members of the aggregate population, and the weakness (raw fitness) of each individual x_i is calculated by summing the strength values of all its dominators. Moreover, an inverse of a k-th nearest neighbor distance (Silverman, 1986) to the rest of the aggregate individuals in the objective space is taken as a local population density measure for each individual x_i (with $k = \sqrt{N_1 + N_2}$, see the procedure below). This value added to the weakness of the same individual x_i gives the estimate of the final (inverse) fitness function being minimized. A suitable truncation mechanism is applied to preserve a constant number of individuals in an external population and to preserve boundary solutions (Zitzler *et al.*, 2001). A very general scheme of the SPEA 2 algorithm is outlined as Procedure 2.4.

Procedure 2.4. SPEA 2

Generate randomly a regular population V1 of N_1 individuals $\{\boldsymbol{x}_i\}_{i=1}^{N_1}$
Create an empty external population V_2 with a size N_2
while $t \leq t\max$
  Assign the fitness value of the individuals in V_1 and V_2
  Copy the non-dominated individuals from V_1 and V_2 to V_2'
  if the size of V_2' exceeds N_2 **then**
    Delete solutions of inferior quality by truncation
    else
    Fill V_2' with dominated individuals of V_1 and V_2
  end if
  Select the parents V_1" from V_2' by the binary tournament selection
  Create the offspring V_1' of V_1" by making:
    Crossover
    Mutation
  Replace the populations: $V_1 \leftarrow V_1'$, $V_2 \leftarrow V_2'$
  Cycle: $t \leftarrow t + 1$
end while (cycling)
end (of finding the solution set V_2)

When solving highly dimensional EMO problems, the conception of the Pareto domination appears to be inefficient. For such cases, Kowalczuk and Białaszewski (2001; 2002; 2003) propose a new **GGA** approach consisting in employing genetic genders for the purpose of making a distinction between different groups of objectives. As appears from the results presented therein, and despite dissimilar basis and judgment, this method also allows keeping diversity among the Pareto optimal solutions produced.

2.3. Pareto Optimization

In the applied ranking methods we avoid the arbitrary weighting of objectives. Instead, a useful classification of the solutions is applied that takes into account particular objectives effectively enough. Their utilitarian representatives are the ranks relating to the Pareto optimality (Goldberg, 1989; Man *et al.*, 1997; Michalewicz, 1996), which allows assessing multi-profit maximization solutions as dominated or non-dominated (P-optimal). The condition of the Pareto optimality for a maximization task on the vector profit function $\boldsymbol{f}(\boldsymbol{x})$ can be precisely formulated as follows (Goldberg, 1989).

Let us consider two solutions that are characterized by the corresponding vectors of profit functions $\boldsymbol{f}(\boldsymbol{x}_r),\ \boldsymbol{f}(\boldsymbol{x}_s) \in \mathbb{R}^M$. The vector $\boldsymbol{f}(\boldsymbol{x}_r)$ is partially smaller than the vector $\boldsymbol{f}(\boldsymbol{x}_s)$ if and only if for all their coordinates the following conditions are fulfilled:

$$f_i(\boldsymbol{x}_r) \leq f_i(\boldsymbol{x}_s), \quad \forall i = 1, 2, \ldots, m, \tag{5}$$

and there is an index i such that $f_i(\boldsymbol{x}_r) < f_i(\boldsymbol{x}_s)$. Thus, in the Pareto sense, a solution $\boldsymbol{x}_r$ is dominated if there exists a solution $\boldsymbol{x}_s$ whose vector of profit functions $\boldsymbol{f}(\boldsymbol{x}_s)$ is *partially better* than $\boldsymbol{f}(\boldsymbol{x}_r)$ in terms of the definition (5). Each *non-dominated* solution $\mathbf{x}_s$ is taken as *Pareto optimal* (*P-optimal*).

Let us consider an assessment of the set of solutions $\{\boldsymbol{x}_1,\ \boldsymbol{x}_2,\ \boldsymbol{x}_3,\ \boldsymbol{x}_4, \boldsymbol{x}_5,\ \boldsymbol{x}_6,\ \boldsymbol{x}_7,\ \boldsymbol{x}_8\}$ in the Pareto sense for a two-dimensional goal vector $\boldsymbol{f}(\boldsymbol{x}) = [\ f_1(\boldsymbol{x}) \quad f_2(\boldsymbol{x})\ ]^{\mathrm{T}}$, as shown in Fig. 1 (Kowalczuk and Białaszewski, 2004a). By applying the Pareto optimality approach, only the solutions $\{\boldsymbol{x}_1,\ \boldsymbol{x}_6,\ \boldsymbol{x}_7,\ \boldsymbol{x}_8\}$, dominating over the corresponding subset of the remaining solutions, are shown to be P-optimal. The P-optimal solutions situated in the dark areas are mutually equivalent. It results from the figure that the solution $\boldsymbol{x}_2$ of the secondary Pareto front is dominated by one solution $\boldsymbol{x}_1$ of the primary P-front, while the solution $\boldsymbol{x}_5$ is dominated by $\boldsymbol{x}_6$. The solution $\boldsymbol{x}_4$ is dominated by two solutions, $\boldsymbol{x}_6$ and $\boldsymbol{x}_5$, whereas $\boldsymbol{x}_3$ is dominated by four solutions, $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, $\boldsymbol{x}_5$ and $\boldsymbol{x}_6$. The solutions $\boldsymbol{x}_7$ and $\boldsymbol{x}_8$, which are neither dominated nor non-dominated, are isolated cases.

Not only does the assessment of solutions concerning their Pareto optimality determine the P-optimal set of solutions, but it also allows useful ranking of all possible solutions with respect to the degree of domination. Namely, each solution can be assigned a certain scalar quantity called a rank (Goldberg, 1989; Man *et al.*, 1997), which can have different definitions, interpretations, and

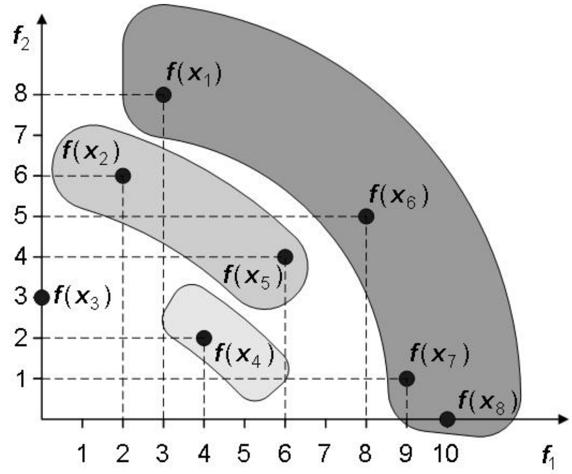


Fig. 1. Illustration of domination in the Pareto sense for two objectives.

applications (Coello, 2001; Horn *et al.*, 1994; Schaffer, 1985; Srinivas and Deb, 1994; Zitzler *et al.*, 2001). In general, though such a rank more or less directly relates to the number of individuals in the current population by which the analyzed individual is dominated (or over which the analyzed individual dominates) in the sense of Pareto (Kowalczuk *et al.*, 1999a; Kowalczuk and Białaszewski, 2004a). Here, the rank $\rho(\boldsymbol{x}_i)$ of a given solution $\boldsymbol{x}_i$ amongst N possible solutions is computed by using the following formula:

$$\begin{aligned} \rho(\boldsymbol{x}_i) &= \mu_{\max} - \mu(\boldsymbol{x}_i) + 1, \\ \mu_{\max} &= \max_{i=1,2,\dots,N} \mu(\boldsymbol{x}_i), \end{aligned} \tag{6}$$

where $\mu(\boldsymbol{x}_i)$ is the degree of domination, meaning the number of solutions by which $\boldsymbol{x}_i$ is dominated, while $\mu_{\max}$ is the maximum value from amongst all $\mu(\boldsymbol{x}_i)$. This kind of ranking transforms the vector of profit functions into a scalar (one-dimensional) space.

According to the above, and as can be seen from Fig. 1, the degrees of domination of the P-optimal solutions are $\mu(\boldsymbol{x}_1) = \mu(\boldsymbol{x}_6) = \mu(\boldsymbol{x}_7) = \mu(\boldsymbol{x}_8) = 0$, because no solution dominates over them. The remaining solutions have the following degrees of domination: $\mu(\boldsymbol{x}_2) = \mu(\boldsymbol{x}_5) = 1$, $\mu(\boldsymbol{x}_4) = 2$, $\mu(\boldsymbol{x}_3) = 4$, giving the maximal degree of domination $\mu_{\max} = \max_{i=1,2,\dots,8} \{\, \mu(\boldsymbol{x}_i) \,\} = 4$.

Finally, the following ranks of the analyzed solutions result from (6): $\rho(\boldsymbol{x}_1) = \rho(\boldsymbol{x}_6) = \rho(\boldsymbol{x}_7) = \rho(\boldsymbol{x}_8) = 5$, $\rho(\boldsymbol{x}_2) = \rho(\boldsymbol{x}_5) = 4$, $\rho(\boldsymbol{x}_4) = 3$, $\rho(\boldsymbol{x}_3) = 1$. The P-optimal solutions $\{\, \boldsymbol{x}_1,\ \boldsymbol{x}_6,\ \boldsymbol{x}_7,\ \boldsymbol{x}_8 \,\}$, having the maximal rank, constitute the primary (highest) Pareto front. The solutions $\{\, \boldsymbol{x}_2,\ \boldsymbol{x}_5 \,\}$ constitute the secondary Pareto front. The solution $\{\, \boldsymbol{x}_4 \,\}$ represents the third Pareto front, while $\{\, \boldsymbol{x}_3 \,\}$, having the minimal rank among all the analyzed solutions, constitutes the fourth (in row) Pareto front, or the fifth with regard to the value of the rank.

2.4. Global Optimality

In the ranking method with respect to the Pareto optimality we effectively transform the profit vector into a scalar value. The approach to optimality does not, however, give any directions as to the choice of a single solution from amongst the Pareto optimal solutions found. Therefore, it is the designer who has to make an independent judgment of the obtained offers.

In order to utilize that freedom, a development of the ranking method was proposed (Kowalczuk and Białaszewski, 2000a; 2004a) that uses the idea of a *global optimality level*. In particular, the vector profit of each solution is mapped to a scalar global optimality level (index), which allows useful ordering of the obtained solutions. Even though there is a chance of obtaining equal indices of global optimality for several solutions constraining the opportunity of obtaining the ideal serial ordering of solutions without additional interference by the designer, this approach effectively limits the number of the equivalent P-optimal solutions.

The method of estimating the global optimality level is given in Procedure 2.5.

Procedure 2.5. Global Optimality (GOL)

Determine a maximal value of each profit function $f_{i_{\max}}$ from amongst all the N solutions (or only the P-optimal ones)

$$\forall_{i=1,2,\dots,m}\ f_{i_{\max}} = \max_{j=1,2,\dots,n} \{\, f_i(\boldsymbol{x}_j) \,\} \tag{7}$$

Assign each solution a global optimality level as

$$\eta_j = \min_{i=1,2,\dots,m} \frac{f_i(\boldsymbol{x}_j)}{f_{i_{\max}}} \tag{8}$$

end (of GOL)

The method of ordering with respect to the global optimality level permits a significant minimization of the problem of the ambiguity of Pareto solutions, which is a cumbersome issue in designs using the P-optimality.

Let us apply Procedure 2.5 to the set of the eight solutions depicted in Fig. 1. The following vector can represent the maximal values of the coordinates of the profit vector: $\boldsymbol{f}_{\max} = [\ f_1(\boldsymbol{x}_8) \quad f_2(\boldsymbol{x}_1)\ ]^{\mathrm{T}} = [\ 10 \quad 8\ ]^{\mathrm{T}}$. In the second step of Procedure 2.5. we obtain the following ordering of the solutions according to the global optimality level η_j $(j = 1, 2, \dots, 8)$: $\boldsymbol{x}_6 \to \eta_6 = 0.6$, $\boldsymbol{x}_2 \to \eta_2 = 0.2$, $\boldsymbol{x}_5 \to \eta_5 = 0.5$, $\boldsymbol{x}_7 \to \eta_7 = 0.1$, $\boldsymbol{x}_1 \to \eta_1 = 0.3$, $\boldsymbol{x}_3,\ \boldsymbol{x}_8 \to \eta_3 = \eta_8 = 0$,

$x_4 \rightarrow \eta_4 = 0.25$. The solution x_6 from the primary Pareto front has a maximal value of the global optimality level equal to 0.6. The worst solutions x_3 (from the last P-front) and x_8 (from the primary Pareto front) have the global optimality level of the null value. The presented ordering method with respect to the global optimality level clearly presents two solutions x_6 and x_5 as most optimal. What is important, the problematic solutions, which maximize only some criteria or one of them (even though they belong to the primary Pareto front as x_7 and x_8), can be easily ruled out in virtue of the global optimality level η. The highest optimality level is then assigned to solely one solution ($x_6 \rightarrow \eta_6 = 0.6$). Thus this technique facilitates a useful determination of one P-optimal solution of the highest global optimality. Though in the example of Fig. 1 the solution x_6 may be intuitively recognized as the best one, in higher dimensional spaces of optimized objectives this need not be that clear.

Within the GA/EC approach, the ordering method according to global optimality can be most successfully applied to the final assessment of a set of P-optimal solutions.

3. Niching Methods

The process of dynamical optimization results from the rivalry between different individuals or species in nature. It raises a hope for the survival of best individuals or species, and the population as a whole. The selected individuals of a high fitness have greater chances of producing offspring of desirable, inherited features. Such a continual selection process leads to individuals of improved suitability. Quite surprisingly, the nature sometimes admits (with a small probability) the survival of individuals weakly fitted. Such individuals establish the source of the diversity of the population, which allows introducing some innovations into the genetic information transmitted to new generations (a kind of stochastic testing non-optimal possibilities). In nature, the time for protecting weak individuals is usually short, and, as a result, they quickly disappear. In order to sustain them, such weak species should be bred under a protective 'umbrella' (Kowalczuk and Białaszewski, 2004a).

In an analogous way, niching in genetic algorithms is a mechanism that preserves not only the best individuals, but also average and weak individuals in order to sustain diverse generations, and that gives a chance to relay the genetic codes of such individuals to offspring (Goldberg, 1989; Kowalczuk *et al.*, 1999a; Kowalczuk and Białaszewski, 2004a). The mechanism balances the populations of the existing species by increasing the chance of mating for the individuals from sparse niches and decreasing that chance for the ones from dense niches. Such a mechanism of 'uniform breeding' also prevents GAs from premature convergence, as well as supports their ability to adapt.

The niche is a finite 'ball' in the space of parameters, in which at least one individual is situated. It is assumed that geometrically close individuals have also similar characteristics with respect to their fitness. Thus they can be identified as species of a distinct sort. The degree of kinship between two individuals can be represented by the closeness function of the geometrical distance between them, which takes its values from the range $[0,\ 1]$, and is also called in the literature (Goldberg, 1989; Michalewicz, 1996) a sharing function. The zero value of the closeness function means that two individuals are not related, i.e. they do not belong to one species, while the unity value denotes their closest relation.

The niche identifying a species of 'bred' individuals is defined as a hyper-ellipsoid in the parameter space. The closeness functions for the individuals/phenotypes x_i and $x_j \in \mathbb{R}^n$, $(i, j = 1, 2, \ldots, N)$, respectively, can be expressed as

$$\delta_{ij} = \begin{cases} 1 - \|x_i - x_j\|_P & \text{if} \quad 0 \leq \|x_i - x_j\|_P < 1, \\ 0 & \text{if} \quad \|x_i - x_j\|_P \geq 1, \end{cases} \tag{9}$$

where

$$\|x_i - x_j\|_P = \sqrt{(x_i - x_j)^{\mathrm{T}} P^{-2} (x_i - x_j)}, \tag{10}$$

$$P = \mathrm{diag}\{\ \phi_1 \quad \phi_2 \quad \ldots \quad \phi_N\ \}, \tag{11}$$

while ϕ_k $(k = 1, 2, \ldots, n)$ is the k-th radius of the hyper-ellipsoid centered on the i-th individual. This radius can be determined as follows:

$$\phi_k = \frac{\Delta_k}{2\varepsilon}, \quad k = 1, 2, \ldots, n, \tag{12}$$

where Δ_k is the width of the real interval of the k-th parameter, while ε represents a real positive factor. The asymptotic case, when $\varepsilon = M \rightarrow \infty$ holds and there are only identical individuals in the niche considered, can be called *auto niching*. By definition, the closeness function has the range $[0,\ 1]$. The zero value of the closeness function means that the two individuals are not related, i.e. they do not belong to the analyzed species. On the other hand, the unity value signifies that the individuals are strongly related or identical.

Once we have defined the niche and the closeness function, the niching technique can be specified as a modification of the magnitude of the fitness-degree vector or the scalar rank (related to the P-optimality) of each individual in its own niche according to the following (Goldberg, 1989; Kowalczuk *et al.*, 1999a; Kowalczuk and Białaszewski, 2004a; Michalewicz, 1996):

$$\tilde{f}(\boldsymbol{x}_i) = \frac{\boldsymbol{f}(\boldsymbol{x}_i)}{\sum_{j=1}^{N} \delta_{ij}} \quad \text{or} \quad \tilde{\rho}(\boldsymbol{x}_i) = \frac{\rho(\boldsymbol{x}_i)}{\sum_{j=1}^{N} \delta_{ij}}, \qquad (13)$$

where $\boldsymbol{f}(\boldsymbol{x}_j)$ is the vector fitness and $\rho(\boldsymbol{x}_i)$ is the scalar rank of the i-th individual, while $\tilde{f}(\boldsymbol{x}_i)$ and $\tilde{\rho}(\boldsymbol{x}_i)$ denote its corresponding niche-adjusted fitness and rank, respectively. The sums in both denominators of (13) concern the set of individuals in the dynamically determined niche centered on the i-th individual/ phenotype. If the individual is the only member of its own niche, then its fitness degree is not decreased, as $\sum \delta_{ij} = 1$. In other cases, the fitness degree is decreased according to the number of neighbors in the niche.

Figure 2 shows the two-dimensional cube of the sought parameters $x_1 \in [\underline{x}_1, \bar{x}_1]$ and $x_2 \in [\underline{x}_2, \bar{x}_2]$. The searched space, defined by the exploration ranges $\Delta_1 = |\bar{x}_1 - \underline{x}_1|$ and $\Delta_2 = |\bar{x}_2 - \underline{x}_2|$, has been divided into equal nine parts because for $\varepsilon = 3$ each individual gets its own niche in the form of an ellipse, shown in Fig. 3, with the diameters $\Delta_1/3$ and $\Delta_2/3$.

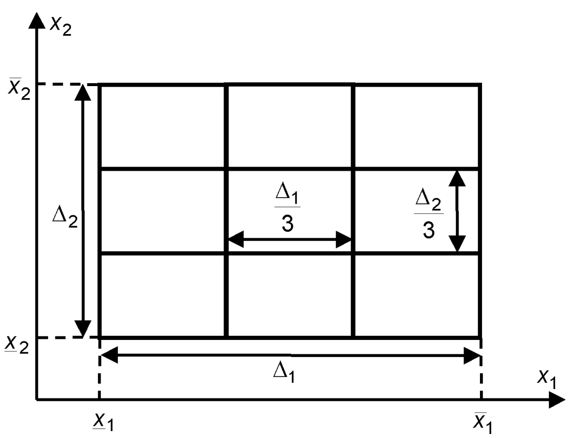


Fig. 2. Division of a two-dimensional domain of the sought parameters.

Consider an exemplary process of niching a two-dimensional vector of fitness. The arrangement of 12 individuals in a 2-dimensional parameter space is given in Fig. 4(a), and their fitness values are shown in Fig. 4(b). The data of a single Pareto optimal solution are marked with a ‘star’, and the niche of it depicted in Fig. 4(a) has the radii 8/6 and 7/6. Figure 4(c) presents the niche-adjusted fitness vectors of individuals prepared for the ranking selection. By niching, the fitness degree of ten individuals (the star and the circles) has been decreased (as the arrows show), and the fitness of two individuals (the cross and one dot) remains intact.

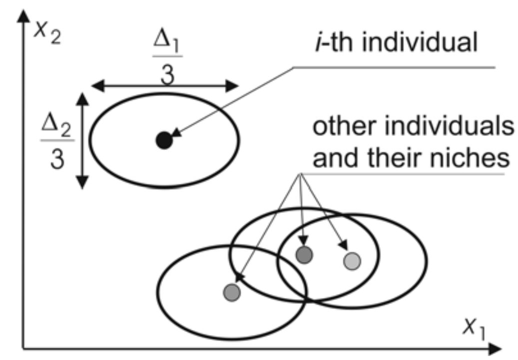


Fig. 3. Ellipsoidal niches of individuals.

As has been described by (13), niching can concern either the fitness or the rank of the individuals of the analyzed population. Procedures 3.1 and 3.2 explain the principles of both algorithms. On the basis of the performed experiments (Kowalczuk and Białaszewski, 2000b; 2004a; 2004b), we can characterize the niching of ranks as a “stronger” mechanism. This effect directly results from the change of the ranks and the domination structure. At the same time, the niching of fitness constitutes solely a source of an indirect modification of the ranks, and changing the fitness vector need not alter its level of domination.

Procedure 3.1. Niching the Fitness (NF)

Compute the fitness of individuals:

$$\boldsymbol{x}_i \rightarrow \boldsymbol{f}(\boldsymbol{x}_i)$$

Warp the fitness of individuals:

$$\boldsymbol{f}(\boldsymbol{x}_i) \rightarrow \tilde{\boldsymbol{f}}(\boldsymbol{x}_i)$$

Assign the ranks of individuals:

$$\tilde{\boldsymbol{f}}(\boldsymbol{x}_i) \rightarrow r(\boldsymbol{x}_i)$$

Select a parental pool based on $r(\boldsymbol{x}_i)$

end (of NF)

Procedure 3.2. Niching the Ranks (NR)

Compute the fitness of individuals:

$$\boldsymbol{x}_i \rightarrow \boldsymbol{f}(\boldsymbol{x}_i)$$

Assign the ranks of individuals:

$$\boldsymbol{f}(\boldsymbol{x}_i) \rightarrow r(\boldsymbol{x}_i)$$

Warp the ranks of individuals:

$$r(\boldsymbol{x}_i) \rightarrow \tilde{r}(\boldsymbol{x}_i)$$

Select a parental pool according to the modified ranks $\tilde{r}(\boldsymbol{x}_i)$

end (of NR)

Analogous types of the niching mechanism can be considered with respect to the parental pool of selected

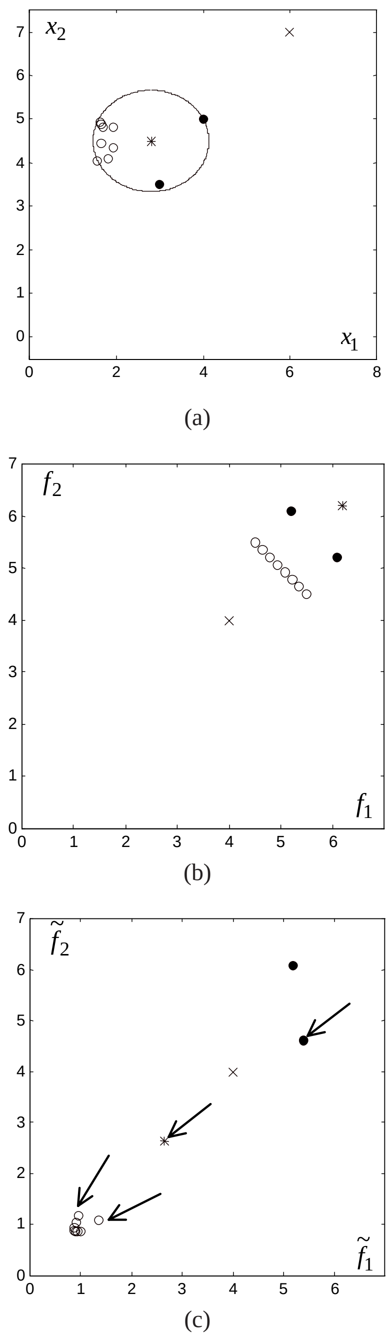


Fig. 4. Effect of exemplary niching: (a) the population and the ellipse niche of the optimal solution; (b) the true fitness amongst the population; (c) the niche-adjusted fitness.

individuals (Kowalczuk and Białaszewski, 2000b; 2004a; 2004b). Procedures 3.3 and 3.4 present the basics of such algorithms.

Procedure 3.3. Niching the Fitness of Parents (NFP)

Compute the fitness of individuals:

$$\boldsymbol{x}_i \rightarrow \boldsymbol{f}(\boldsymbol{x}_i)$$

Assign the ranks of individuals:

$$\boldsymbol{f}(\boldsymbol{x}_i) \rightarrow r(\boldsymbol{x}_i)$$

Select a parental pool according to the ranks $r(\boldsymbol{x}_i)$

Warp the fitness of parents:

$$\boldsymbol{f}(\boldsymbol{x}_j) \rightarrow \tilde{f}(\boldsymbol{x}_j)$$

Assign the ranks of parents:

$$\tilde{f}(\boldsymbol{x}_j) \rightarrow r(\boldsymbol{x}_j)$$

Select a final parental pool according to the ranks $r(\boldsymbol{x}_j)$

end (of NFP)

Procedure 3.4. Niching of the Ranks of Parents (NRP)

Compute the fitness of individuals:

$$\boldsymbol{x}_i \rightarrow \boldsymbol{f}(\boldsymbol{x}_i)$$

Assign the ranks of individuals:

$$\boldsymbol{f}(\boldsymbol{x}_i) \rightarrow r(\boldsymbol{x}_i)$$

Select a parental pool according to the ranks

$$r(\boldsymbol{x}_i)$$

Warp the ranks of parents:

$$r(\boldsymbol{x}_j) \rightarrow \tilde{r}(\boldsymbol{x}_j)$$

Select a final parental pool according to the modified ranks $\tilde{r}(\boldsymbol{x}_j)$

end (of NRP)

Taking the survival of an individual as an appearance in its genetic material of the next generation, niching allows both increasing the probability of survival for species of sparse niches and decreasing it for those of dense niches. This means that this mechanism has the nature of ‘uniform breeding’. It is important that in spite of such a uniform breeding policy, a global effect of genetic expansion and selection procedures can be observed, which consists in constant densities persistently sustained in certain niches. This can be eventually interpreted in terms of their robustness to changes in the fitness measure.

The niching mechanism should also be studied in terms of time conditioning within GA cycles. Therefore, taking into account the fact that niching is useful (Kowalczuk and Białaszewski, 2000b; 2004a; 2004b) though time consuming, we propose considering periodic niching as an alternative to the (classical) continuous one. Within the periodic mechanism, by definition, niching is limited to a certain number of consecutive generations (cycles), and then it is switched off until the subsequent period.

4. Illustrative Example

All the niching methods considered have been applied to exemplary multi-objective optimization tasks (Kowalczuk and Białaszewski, 2000b). For simplicity, let us first take a two-objective function:

$$\min_{x} \boldsymbol{f}(\boldsymbol{x}) = \min_{x} \big[f_1(\boldsymbol{x}) \ \ f_2(\boldsymbol{x}) \big], \tag{14}$$

where the compound functions are

$$\begin{aligned} f_1(\mathbf{x}) &= \varphi_1(x_1,\ x_2) \\ &= -\sum_{j=1}^{10} d_j \left[\sum_{i=1}^{2} (x_i - a_{ij})^2 + c_j \right]^{-1}, \end{aligned} \tag{15}$$

$$\begin{aligned} f_2(\mathbf{x}) &= \varphi_2(x_1,\ x_2) \\ &= -\sum_{j=1}^{10} e_j \left[\sum_{i=1}^{2} (x_i - b_{ij})^2 + c_j \right]^{-1}, \end{aligned} \tag{16}$$

and their coefficients have the values given in Table 1.

Table 1. Coefficients of the functions (15) and (16).

j	a_{1j}	a_{2j}	b_{1j}	b_{2j}	c_j	d_j	e_j
1	4	4	8	4.6	0.1	1	1
2	1	1	5	1	0.2	1	1.5
3	8	8	9	2	0.2	1.5	1
4	6	6	5	5	0.4	1	0.5
5	3	7	3	10	0.6	1	1
6	2	9	1	6	0.6	1	1
7	5	5	6	6	0.3	1.2	0.5
8	8	1	7	7	0.7	2	4
9	6	2	2	2	0.5	1	1
10	7	3.6	5	5	0.5	1	1

The GA used in simulations has the structure described by Procedure 4.1.

Procedure 4.1. General GA scheme
Initiation of individuals population
  while $t \le t\max$
    Fitness assessment
    Niching
    Selection of parents
    Creation of new individuals by:
      – one-point crossover
      – binary mutation
    Setting the new population
    $t = t + 1$
  end while (cycling)
end (of GA)

Setting the new generation consists in exchanging 90% of old individuals with new ones and retaining the best 10% (elite) of the old population. The retained solutions are selected from the Pareto optimal ones ordered based on the global optimality level η.

The simulations were carried out with the use of the following GA parameters: 50 individuals in the population, 32-bits genotypes (binary codes of the phenotypes), the probability of one-point crossover equal to 0.7, the value of mutation probability being 0.02. The initial population was placed at the lower right corner of the searched space illustrated in Fig. 5(d).

Procedure 4.2 describes the applied selection method based on the concepts of proportionality supplemented with the stochastic reminder choice and the 'roulette wheel' (RW) (Kowalczuk and Białaszewski, 2004a).

Procedure 4.2. Evolutionary selection
  Assign the expected number of copies

$$e(\boldsymbol{x}_i) = N\rho(\boldsymbol{x}_i) \Big/ \sum_{i=1}^{N} \rho(\boldsymbol{x}_i), \quad i = 1, 2, \dots, N$$

  Copy $N_{\text{int}} = \sum_{i=1}^{n} \lfloor e(\mathbf{x}_i) \rfloor$ individuals
  Define the number of vacancies $\tilde{N} = N - N_{\text{int}}$
  Assign the distribution functions

$$q(\mathbf{x}_i) = \sum_{j=1}^{i} \{\, e(\mathbf{x}_j) - \lfloor e(\mathbf{x}_j) \rfloor \,\}$$

  Perform $\tilde{N}$ turnings of the RW by:
   – generating a random number $r \in [0,\ 1]$
   – selecting such an individual $\boldsymbol{x}_i$ that

$$r \le \frac{q(\boldsymbol{x}_i)}{q(\boldsymbol{x}_n)}$$

   – copying $\boldsymbol{x}_i$ to the parental pool
end (of selection)

4.1. No Niching

Within the first approach, genetic optimization was carried out without niching (WN). Figure 5(a) presents a histogram of the individuals in particular ranks and in consecutive generations, where a large number (about 60%) of non-dominated individuals can be seen in each generation (starting from the 50-th generation). Figure 5(b) depicts the best fitness obtained in running generations, precisely speaking: the vertices (dots) of the graph represent the vector-valued fitness of the Pareto optimal solutions. Figure 5(c) shows the range set of the fitness function obtained by scanning the whole domain (all possible solutions). Within this area, the final Pareto optimal individuals are represented by the dots, while the crosses mark the remaining dominated solutions.

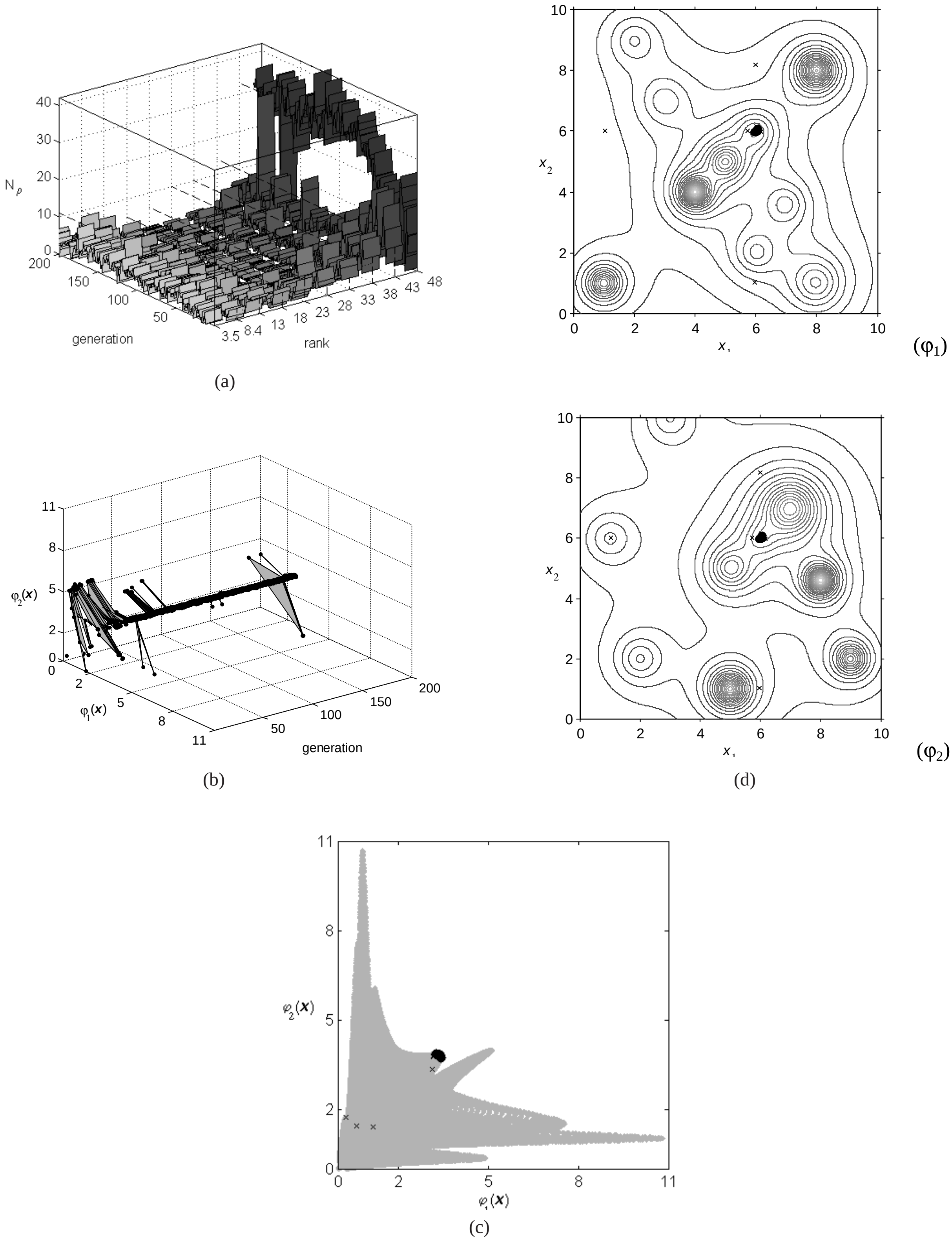


Fig. 5. Results for no niching: the running histogram of the number of individuals in ranks and in generations for the WN evolution (a), the fitness of P-optimal solutions in GA (WN) runs (b), the available fitness area (co-domain) with the fitness of the P-optimal (•) and dominated (×) solutions (WN) (c), the obtained fitness ϕ_1 and ϕ_2 in the searched space found without niching (WN) (d).

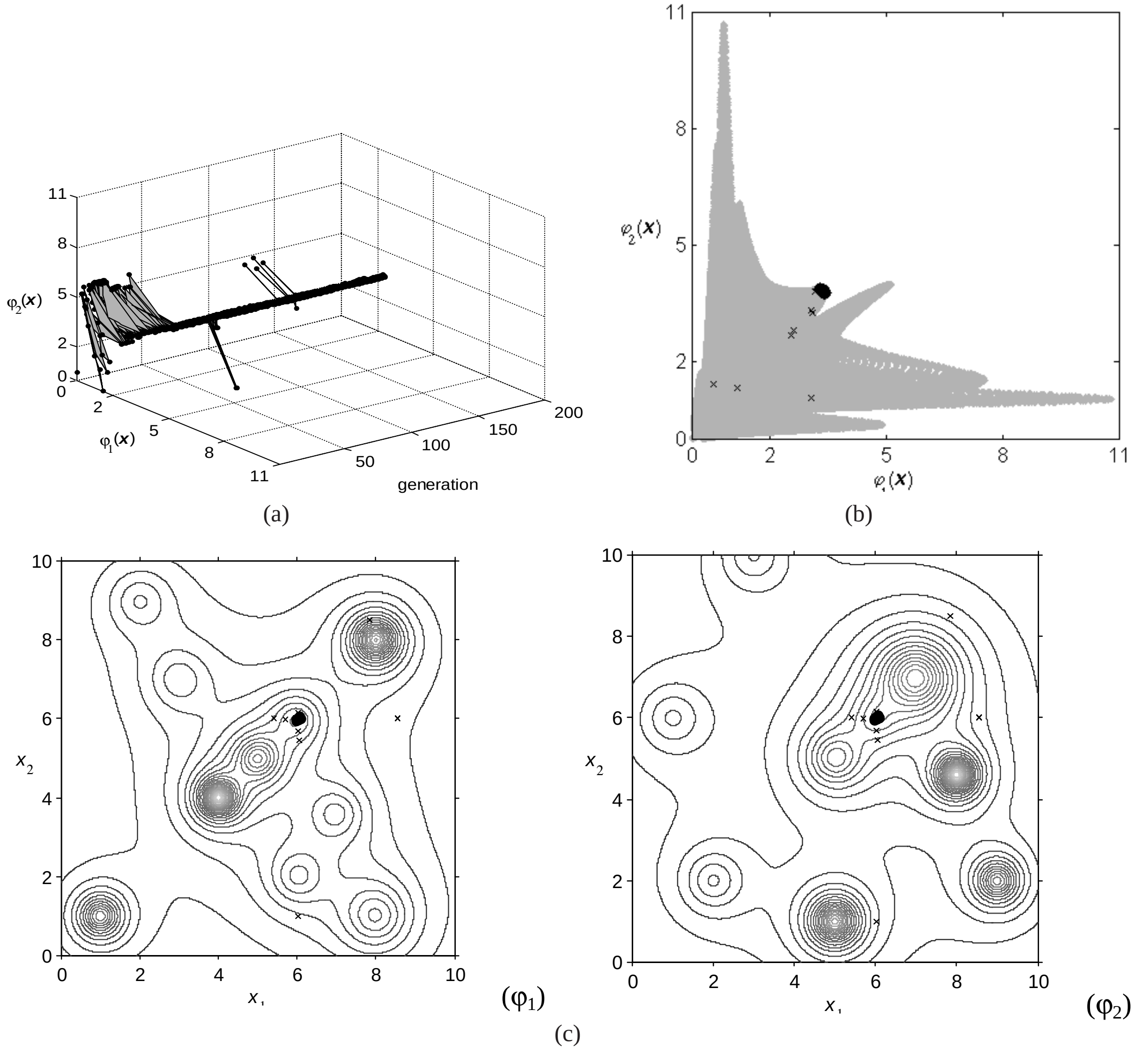


Fig. 6. Results for the niching of the fitness: fitness of P-optimal solutions in GA-NF runs (a), NF solutions in the objective space (b), NF solutions in the searched space (c).

Figure 5(d) illustrates the fitness functions projected on the two-dimensional search space with the non-dominated (dots) and dominated (crosses) individuals marked. As is shown in Figs. 5(b)–(d), the genetic algorithm without niching swiftly tends to sub-Pareto optimal solutions only. In effect, the population is not diverse enough (Fig. 5(d)), and the GA easily overlooks the solutions which belong to a possibly highest Pareto front (Fig. 5(c)).

4.2. Niching

The GA with the niching of the fitness (NF shown in Figs. 6(a)–(c)) and the ranks (NR in Figs. 7(a)–(c)) of individuals has finished the search with only sub-Pareto optimal solutions, though with better diversity.

In contrast to the above, with the mechanism of niching the ranks (NRP given in Figs. 8(a)–(d)) and the fitness (NFP shown in Figs. 9(a)–(d)) performed with respect to the members of the parental pool, the genetic algorithm is able to find those solutions which are totally Pareto optimal, i.e., belong to the highest available Pareto front. At the same time, the two NFP and NRP methods effectively sustain the diversity of evolving populations.

Table 2 contains a brief summary of the results obtained for niching applied using different diameters of niches. It can be seen that reasonably small radii of species niches leads to great diversity in the population and a high number of Pareto optimal solutions, without increasing the complexity of the algorithm.

5. Synthesis of State Observers

The systems of Fault Detection and Isolation (FDI) are used for practical diagnostic purposes. Such systems are

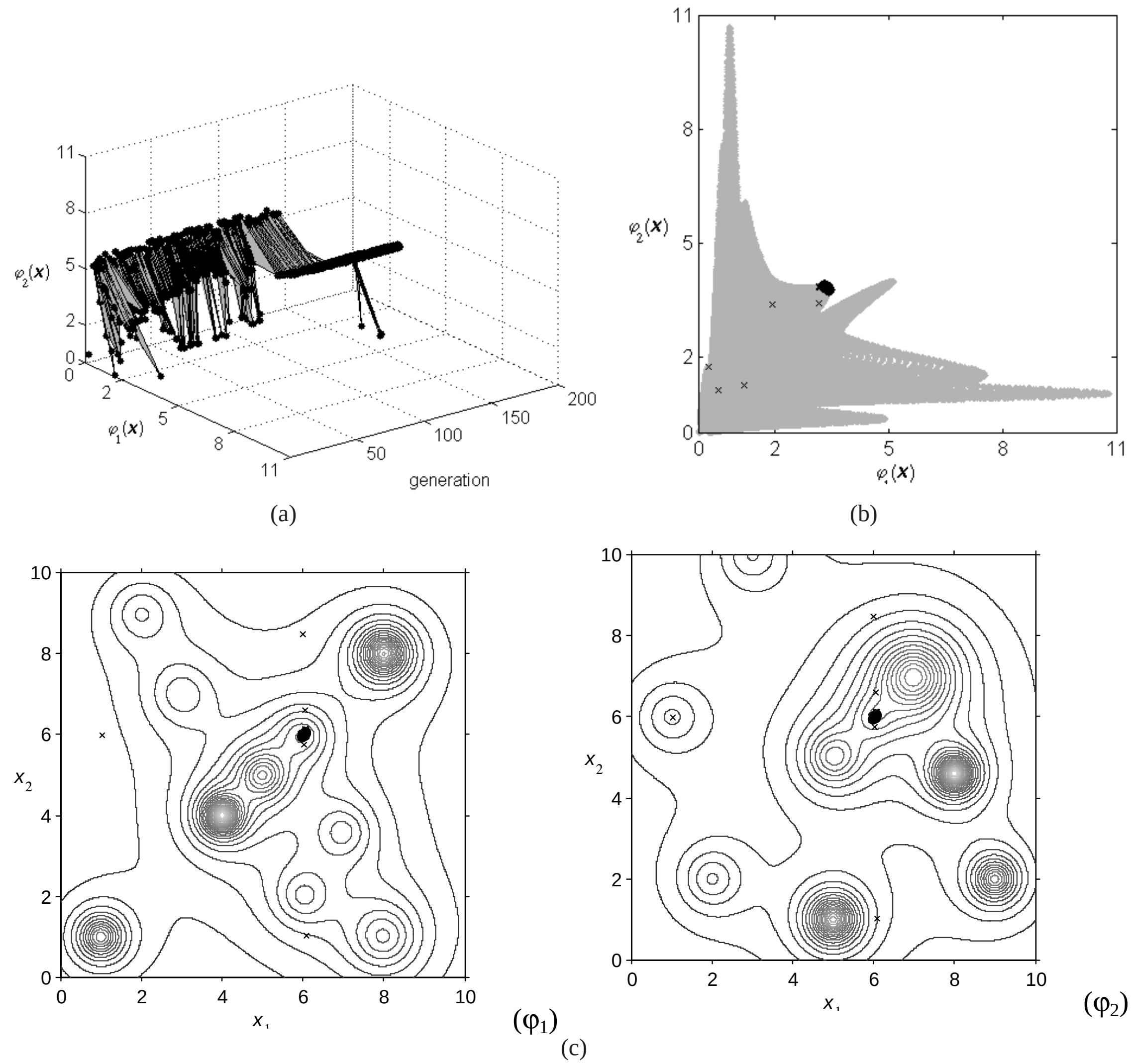


Fig. 7. Results for the niching of the ranks: the fitness of P-optimal solutions in GA-NR runs (a), NR solutions in the co-domain of objectives (b), NR solutions in the space of the sought parameters shown against objectives (c).

built on two principal operations. The first one consists in detecting the occurrence of a fault, and the second one tends to isolate a particular defect from others. The main task of FDI systems is to ensure an expected operation of engineering assemblages used for signal measurement, system monitoring and control. This is of importance in systems of high safety (Chen *et al.*, 1996; Korbicz *et al.*, 2004; Kowalczuk and Białaszewski, 2004a; Patton *et al.*, 1989). Sometimes even a small system error can have a serious effect on system performance. Therefore, FDI should be done as early as possible, so as to allow the supervising operator to take appropriate steps.

The idea of FDI consists in comparing measurements of the plant with signals predicted based on the object's model. Differences between the corresponding signals, called residues or residuals, allow identifying the existing faults of the system. Those differences are, in general, influenced by disturbances, noise and modeling errors. Fault detection is achieved by filtering these residues, and diagnostic decisions are made on the basis of their appropriate evaluation. The scheme of a diagnosis system founded on a state observer and an additional filter, referred to as a *residue generator*, is illustrated in Fig. 10.

Let us consider the following model:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t)+\boldsymbol{B}\boldsymbol{u}(t)+\boldsymbol{N}\boldsymbol{d}(t)+\boldsymbol{F}_1\boldsymbol{f}(t)+\boldsymbol{w}(t), \quad (17)$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) + \boldsymbol{F}_2\boldsymbol{f}(t) + \boldsymbol{v}(t), \quad (18)$$

where $\boldsymbol{x}(t) \in \mathbb{R}^n$ denotes a state vector, $\boldsymbol{u}(t) \in \mathbb{R}^p$ is a control vector, $\boldsymbol{y}(t) \in \mathbb{R}^m$ stands for a measurement vector, $\boldsymbol{f}(t) \in \mathbb{R}^q$ models a fault vector, $\boldsymbol{d}(t) \in \mathbb{R}^n$ is a state disturbance vector, while the signals $\boldsymbol{w}(t) \in \mathbb{R}^n$ and

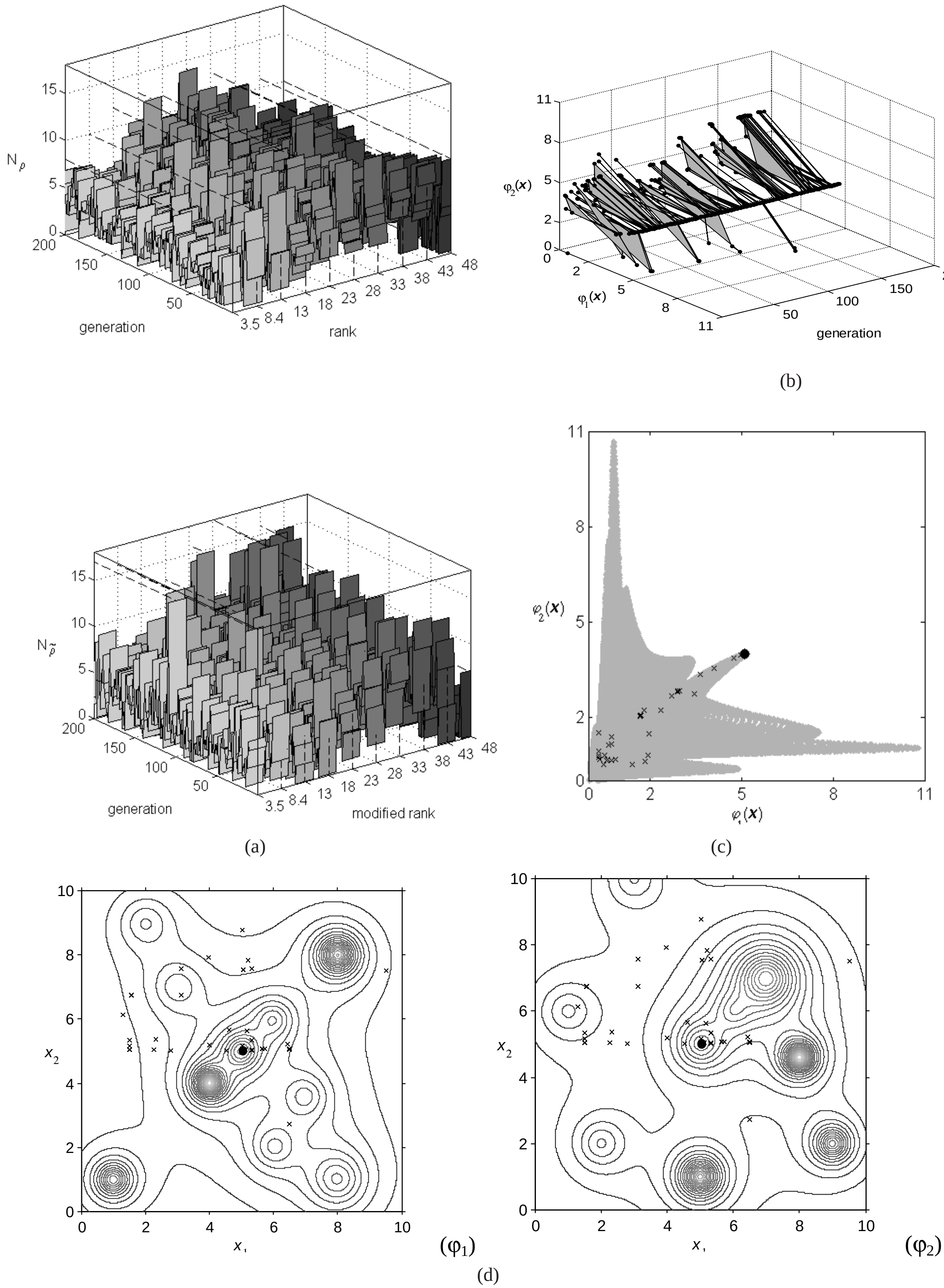


Fig. 8. NFP results: running histograms of the number of solutions in ranks and generations for the NFP type of niching (a), the fitness of P-optimal solutions in GA-NFP runs (b), NFP solutions with respect to the available fitness (c), NFP solutions in the searched space in terms of the two objectives (d).

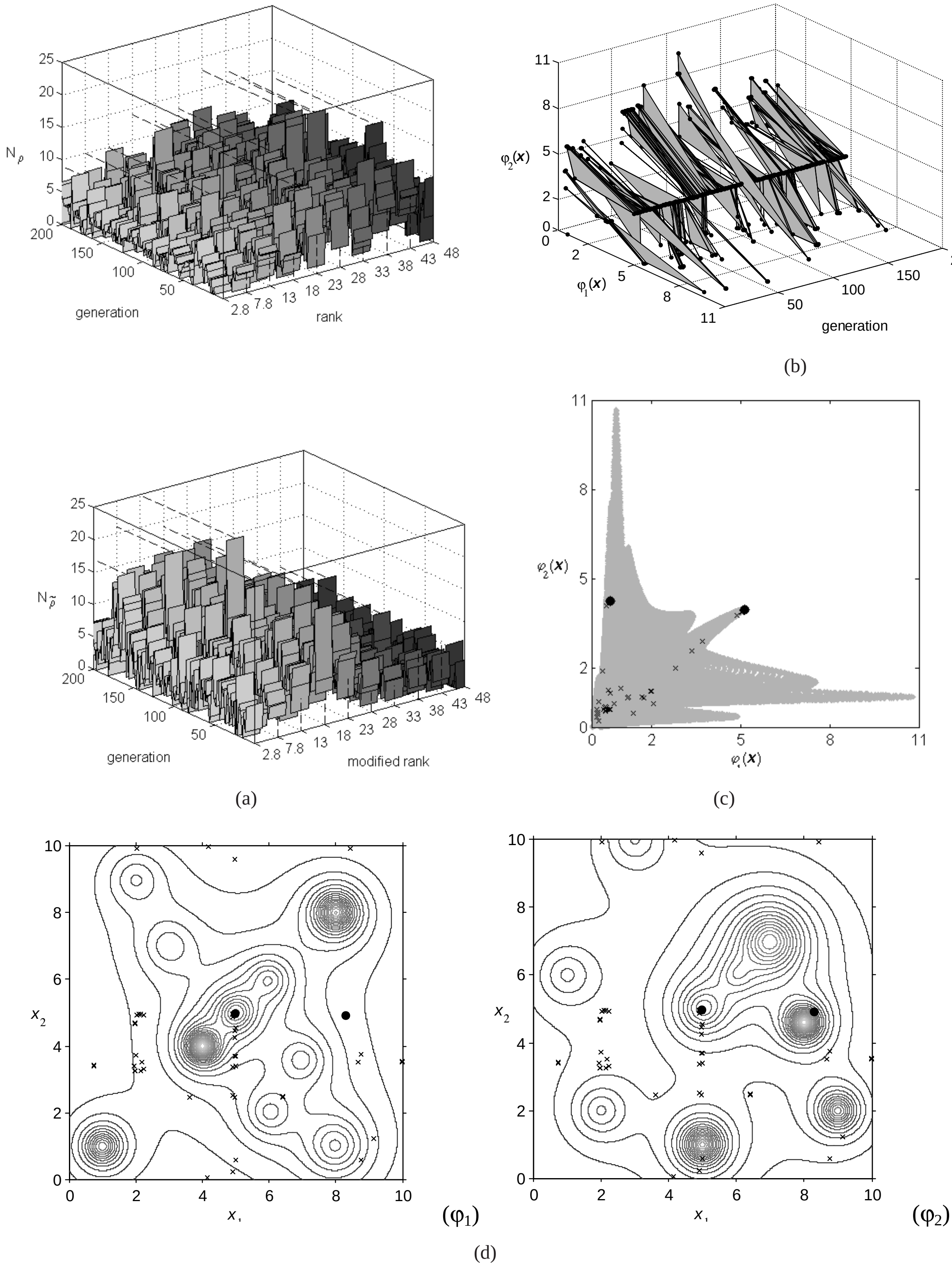


Fig. 9. NRP solutions: running histograms of the number of solutions in ranks and generations for the NRP niching (a), the fitness of Pareto solutions in GA-NRP runs (b), NRP solutions in the space of the optimized fitness (c), NRP solutions in the searched space in terms of objectives (d).

Table 2. Optimality of solutions, diversity of population, and degree of rank warping (or uniform breeding).

$2\phi_k$	NF	NR	NFP	NRP
10^{-1}	1 optimal solution	2 optimal solutions	2 optimal solutions	4 optimal solutions
	small diversity	small diversity	great diversity	greatest diversity
	moderate warping	strong warping	uniform breeding	very strong warping
2	3 optimal solutions	2 optimal solutions	1 optimal solution	1 optimal solution
	great diversity	great diversity	small diversity	small diversity
	moderate warping	strong warping	uniform breeding	very strong warping
10	3 sub-optimal solutions	2 sub-optimal solutions	2 optimal solutions	2 optimal solutions
	small diversity	small diversity	great diversity	great diversity
	reverse warping	very strong warping	reverse warping	very strong warping

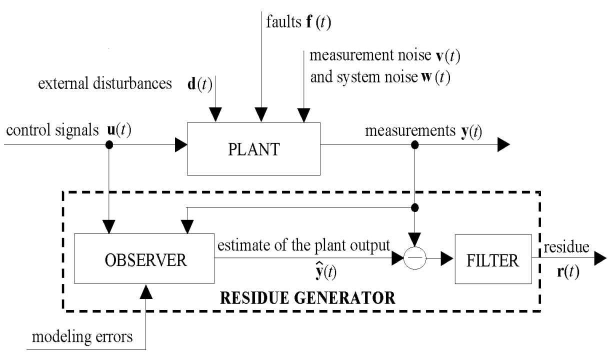


Fig. 10. Scheme of observer-based FDI systems.

$\boldsymbol{v}(t) \in \mathbb{R}^m$ have a noisy character. The matrices of the model (17)–(18) have suitable dimensions: $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, $\boldsymbol{B} \in \mathbb{R}^{n\times p}$, $\boldsymbol{C} \in \mathbb{R}^{m\times n}$, $\boldsymbol{D} \in \mathbb{R}^{m\times p}$, $\boldsymbol{F}_1 \in \mathbb{R}^{n\times q}$ and $\boldsymbol{F}_2 \in \mathbb{R}^{m\times q}$. We assume that the pair $(\boldsymbol{A}, \boldsymbol{C})$ is completely observable, and that the fault $\boldsymbol{f}(t)$ is represented by an unknown vector time function while the influence of this fault on the state evolution of the system considered and on the measurement signals is conditioned by the choice of $\boldsymbol{F}_1$ and $\boldsymbol{F}_2$, respectively.

The state observer can be described by the following equations (Brogan, 1991; Chen *et al.*, 1996; Kowalczuk and Suchomski, 2004a; 2004b; Suchomski and Kowalczuk, 2004):

$$\dot{\hat{\boldsymbol{x}}}(t) = (\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C})\hat{\boldsymbol{x}}(t)+(\boldsymbol{B}-\boldsymbol{K}\boldsymbol{D})\boldsymbol{u}(t)+\boldsymbol{K}\boldsymbol{y}(t), \quad (19)$$

$$\hat{\boldsymbol{y}}(t) = \boldsymbol{C}\hat{\boldsymbol{x}}(t) + \boldsymbol{D}\boldsymbol{u}(t), \quad (20)$$

where $\hat{\boldsymbol{x}}(t) \in \mathbb{R}^n$ is a state-vector estimation, $\hat{\boldsymbol{y}}(t) \in \mathbb{R}^m$ constitutes an estimated system output, while $\boldsymbol{K} \in \mathbb{R}^{n\times m}$ stands for a matrix observer gain.

The residual signal $\boldsymbol{r}(t) \in \mathbb{R}^r$ can be obtained from the following residual equation:

$$\boldsymbol{r}(t) = \boldsymbol{Q}\big(\boldsymbol{y}(t) - \hat{\boldsymbol{y}}(t)\big), \quad (21)$$

where a matrix $\boldsymbol{Q} \in \mathbb{R}^{r\times m}$ of weights serves as a tuning design parameter. The residue generator described by (19)–(21), used for detecting faults in the plant modeled by (17), (18), is presented in Fig. 11.

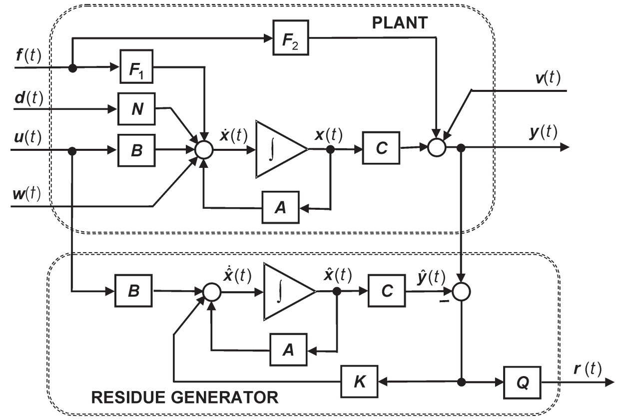


Fig. 11. Residual generator based on the state space model.

The evolution of the state estimation error

$$\boldsymbol{e}(t) = \boldsymbol{x}(t) - \hat{\boldsymbol{x}}(t), \quad \boldsymbol{e}(t) \in \mathbb{R}^n, \quad (22)$$

can be described by the following equation of an *internal form*, conditioned by faults and disturbances:

$$\dot{\boldsymbol{e}}(t) = (\boldsymbol{A} - \boldsymbol{K}\boldsymbol{C})\boldsymbol{e}(t) + (\boldsymbol{F}_1 - \boldsymbol{K}\boldsymbol{F}_2)\boldsymbol{f}(t) + \boldsymbol{N}\boldsymbol{d}(t) + \boldsymbol{w}(t) - \boldsymbol{K}\boldsymbol{v}(t). \quad (23)$$

For an asymptotically stable homogeneous error equation, all eigenvalues of $\boldsymbol{A} - \boldsymbol{K}\boldsymbol{C}$ must have negative real parts. It can be easily shown (Brogan, 1991; Chen *et al.*, 1996; Kowalczuk and Suchomski, 2004a) that the residual vector $\boldsymbol{r}(t)$ of (21) can be interpreted as the observation of the state estimation error $\boldsymbol{e}(t)$ in the presence of the perturbing signals $\boldsymbol{f}(t)$ and $\boldsymbol{v}(t)$, which can be expressed as

$$\boldsymbol{r}(t) = \boldsymbol{Q}\boldsymbol{C}\boldsymbol{e}(t) + \boldsymbol{Q}\boldsymbol{F}_2\boldsymbol{f}(t) + \boldsymbol{Q}\boldsymbol{v}(t). \quad (24)$$

The solution of (23) can be shown in the s-domain:

$$\boldsymbol{E}(s) = \big[s\boldsymbol{I}_n-(\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C})\big]^{-1}\big\{(\boldsymbol{F}_1-\boldsymbol{K}\boldsymbol{F}_2)\boldsymbol{F}(s) + \boldsymbol{N}\boldsymbol{D}(s)+\boldsymbol{W}(s)+\boldsymbol{K}\boldsymbol{V}(s)+\boldsymbol{e}(0)\big\}, \quad (25)$$

where $\boldsymbol{F}(s)$, $\boldsymbol{D}(s)$, $\boldsymbol{W}(s)$ and $\boldsymbol{V}(s)$ are Laplace transforms of the corresponding signals, while $\boldsymbol{e}(0)$ denotes an initial value of the state estimation error. The residue has the following Laplace form (Brogan, 1991; Chen *et* *al.*, 1996; Kowalczuk *et al.*, 1999a; Kowalczuk and Białaszewski, 2004a; Kowalczuk and Suchomski, 2004a; 2004b; Suchomski and Kowalczuk, 2004):

$$\boldsymbol{R}(s) = \boldsymbol{G}_{rf}(s)\,\boldsymbol{F}(s)+\boldsymbol{G}_{rd}(s)\,\boldsymbol{D}(s)+\boldsymbol{G}_{rw}(s)\,\boldsymbol{W}(s) + \boldsymbol{G}_{rv}(s)\,\boldsymbol{V}(s)+\boldsymbol{G}_{re}(s)\,\boldsymbol{e}(0), \tag{26}$$

where the matrix transfer functions are as follows:

$$\boldsymbol{G}_{rf}(s) = \boldsymbol{Q}\left\{\boldsymbol{C}\left[s\boldsymbol{I}_n - (\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C})\right]^{-1} \times (\boldsymbol{F}_1 - \boldsymbol{K}\boldsymbol{F}_2) + \boldsymbol{F}_2\right\}, \tag{27}$$

$$\boldsymbol{G}_{rd}(s) = \boldsymbol{Q}\boldsymbol{C}\big[s\boldsymbol{I}_n - (\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C})\big]^{-1}\boldsymbol{N}, \tag{28}$$

$$\boldsymbol{G}_{rw}(s) = \boldsymbol{Q}\boldsymbol{C}\left[s\boldsymbol{I}_n - (\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C})\right]^{-1}, \tag{29}$$

$$\boldsymbol{G}_{rv}(s) = \boldsymbol{Q}\left\{\boldsymbol{I}_m - \boldsymbol{C}\left[s\boldsymbol{I}_n - (\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C})\right]^{-1}\boldsymbol{K}\right\}, \tag{30}$$

$$\boldsymbol{G}_{re}(s) = \boldsymbol{Q}\boldsymbol{C}\left[s\boldsymbol{I}_n - (\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C})\right]^{-1} = \boldsymbol{G}_{rw}(s). \tag{31}$$

The above transfer functions give a description of the effect of critical factors of faults, initial conditions, external disturbances, and modeling uncertainty.

The matrices $\boldsymbol{K}$ and $\boldsymbol{Q}$ form the parameterization of the designed detector. It is thus necessary to choose the entries of those matrices such that they will emphasize the influence of $\boldsymbol{F}(s)$ on $\boldsymbol{R}(s)$ and, at the same time, restrict the impact of the remaining factors on $\boldsymbol{R}(s)$.

To define such tasks of parametric optimization, let us consider the following weighted partial-objective functions in the whole frequency domain:

$$J_1(\boldsymbol{K},\boldsymbol{Q}) = ||\boldsymbol{W}_1(s)\cdot\boldsymbol{G}_{rf}(s)||_\infty, \tag{32}$$

$$J_2(\boldsymbol{K},\boldsymbol{Q}) = ||\boldsymbol{W}_2(s)\cdot\boldsymbol{G}_{rd}(s)||_\infty, \tag{33}$$

$$J_3(\boldsymbol{K},\boldsymbol{Q}) = ||\boldsymbol{W}_3(s)\cdot\boldsymbol{G}_{rw}(s)||_\infty, \tag{34}$$

$$J_4(\boldsymbol{K},\boldsymbol{Q}) = ||\boldsymbol{W}_4(s)\cdot\boldsymbol{G}_{rv}(s)||_\infty, \tag{35}$$

$$J_5(\boldsymbol{K}) = ||(\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C})^{-1}||_s, \tag{36}$$

$$J_6(\boldsymbol{K}) = ||(\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C})^{-1}\boldsymbol{K}||_s, \tag{37}$$

with the following matrix norms:

$$||\boldsymbol{M}(s)||_\infty = \sup_{\omega}\bar{\sigma}\big[\boldsymbol{M}(j\omega)\big], \tag{38}$$

$$||\boldsymbol{M}||_s = \bar{\sigma}\left[\boldsymbol{M}\right], \tag{39}$$

where $\bar{\sigma}[\cdot]$ is the maximum singular value of the matrix argument of this operator.

The weighting matrix functions $\boldsymbol{W}_1(s)$, $\boldsymbol{W}_2(s)$, $\boldsymbol{W}_3(s)$ and $\boldsymbol{W}_4(s)$, which represent the prior knowledge about the spectral properties of the process, introduce additional degrees of freedom of the detector design procedure. Those matrices permit a spectral separation of the effects of faults and noise. In order to maximize the influence of faults at low frequencies and minimize the noise effect at high frequencies, the matrix function $\boldsymbol{W}_1(s)$ should have a low-pass property. The weighting function $\boldsymbol{W}_2(s)$ should have the same properties, while the spectral effect of $\boldsymbol{W}_3(s)$ and $\boldsymbol{W}_4(s)$ should be opposite to that of $\boldsymbol{W}_1(s)$.

Once we have fixed the weighting matrices $\boldsymbol{W}_1(s)$, $\boldsymbol{W}_2(s)$, $\boldsymbol{W}_3(s)$ and $\boldsymbol{W}_4(s)$, the synthesis of the detection filter reduces to the issue of the multi-objective optimization of the pair $(\boldsymbol{K},\boldsymbol{Q}) \in \mathbb{R}^{n\times m}\times\mathbb{R}^{r\times m}$ with regard to the goal expressed by

$$\underset{(\boldsymbol{K},\boldsymbol{Q})}{\text{opt}}\ \boldsymbol{J}(\boldsymbol{K},\boldsymbol{Q}) = \begin{bmatrix} \max\limits_{(\boldsymbol{K},\boldsymbol{Q})} J_1(\boldsymbol{K},\boldsymbol{Q}) \\ \min\limits_{(\boldsymbol{K},\boldsymbol{Q})} J_2(\boldsymbol{K},\boldsymbol{Q}) \\ \min\limits_{(\boldsymbol{K},\boldsymbol{Q})} J_3(\boldsymbol{K},\boldsymbol{Q}) \\ \min\limits_{(\boldsymbol{K},\boldsymbol{Q})} J_4(\boldsymbol{K},\mathbf{Q}) \\ \min\limits_{\boldsymbol{K}} J_5(\boldsymbol{K}) \\ \min\limits_{\boldsymbol{K}} J_6(\boldsymbol{K}) \end{bmatrix}. \tag{40}$$

With stable $\mathrm{spectr}[\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C}] \subset \wp_-$, a set of complex values with a negative real part, the inverse matrix $[\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C}]^{-1}$ exists. The profit index $J_1(\boldsymbol{K},\boldsymbol{Q})$ constitutes the main maximized criterion (implying a similar positive effect on lower bounds on the minimum singular values), while the cost functions $J_2(\boldsymbol{K},\boldsymbol{Q})$, $J_3(\boldsymbol{K},\boldsymbol{Q})$ and $J_4(\boldsymbol{K},\boldsymbol{Q})$ account for the model disturbance and noise effects. The costs $J_5(\boldsymbol{K})$ and $J_6(\boldsymbol{K})$, describing the influence of static deviations from the nominal model of the plant, represent explicit robustness measures.

The selection of the observer gain $\boldsymbol{K}$ can be performed in several ways. The method of the eigenstructure assignment of the observation system matrix $(\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C})$ or a method based on the Kalman-Bucy filtering can be applied, for instance (in the latter case, the knowledge of covariance characteristics of noise perturbations in the analyzed model is necessary). Here we follow the first approach (Chen *et al.*, 1996), in which a whole spectrum (eigenvalues λ_i) of the observation system $(\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C})$ is placed in the required region while assuring the necessary robustness of this placement to the deviations $(\Delta\boldsymbol{A},\Delta\boldsymbol{C})$ from the nominal plant model.

It is important to emphasize that in an 'original' FDI design problem the issue of structural synthesis is not complete – in this sense that only a part of design freedom within the matrix $\boldsymbol{K}$ is utilized during the design. Therefore, the spectral synthesis of the matrix $(\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C})$

should consider an additional task of robust stabilization of the observer by considering a suitably parameterized family of pairs $\{(\boldsymbol{A}+\Delta\boldsymbol{A}, \boldsymbol{C}+\Delta\boldsymbol{C})\}$ that map the uncertainty of modeling (Kowalczuk and Suchomski, 2004b).

Chen *et al.* (1996) utilized the method of sequential inequalities (Zakian and Al-Naib, 1973) in a GA optimization procedure. In their approach, cost indices are expressed in the frequency domain and transformed into a set of inequality constraints, which are tested for a finite set of frequencies. Thus the authors applied a GA to search for optimal solutions satisfying all inequality constraints. To get a suitable parameterization of the gain $\boldsymbol{K}$, Chen *et al.* (1996) employed the eigenstructure assignment method.

In our approach the analyzed multi-objective optimization problem is solved by a method that incorporates both the Pareto optimality and the genetic search in the whole frequency domain. In particular, the design of residue generators is based on the optimization of the objectives $\boldsymbol{J}(\boldsymbol{K},\boldsymbol{Q})$ of (40), whose coordinates are partial objectives: the profit function $J_1(\boldsymbol{K},\boldsymbol{Q})$ and the cost functions $J_i(\boldsymbol{K},\boldsymbol{Q}),\ i = 2,\ 3,\ 4,\ 5, 6$. The ranking method derived from the P-optimality is employed to assess the P-optimal solutions of this task generated by the genetic algorithm operating on multi-allele codes.

To assure that genetic optimization yields exclusively permissible solutions: $\operatorname{spectr}[\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C}] \subset \wp$, we directly search only for eigenvalues (and not for the observer gain $\boldsymbol{K}$ itself), on the basis of which the observer gain matrix is calculated by means of the *pole placement* method (Kowalczuk *et al.*, 1999a; Kowalczuk and Białaszewski, 2004a).

The problem of multi-objective optimization is thus reduced to the following task:

$$\underset{(\boldsymbol{K},\boldsymbol{Q})}{\operatorname{opt}}\ \boldsymbol{J}(\boldsymbol{K},\boldsymbol{Q}) = \underset{\boldsymbol{K}}{\operatorname{opt}}\ \boldsymbol{J}(\boldsymbol{K}) = \underset{\lambda}{\operatorname{opt}}\ \boldsymbol{J}\big(\boldsymbol{K}(\lambda)\big), \tag{41}$$

where $\lambda \subset \wp^N$ is the sought n-element vector of the eigenvalues of the matrix $(\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C})$.

5.1. Ship Propulsion System

The ship propulsion system of a low-speed marine vehicle (Izadi and Blanke, 1998; Kowalczuk and Białaszewski, 2000a; 2003) that consists of one engine and one propeller is taken as a practical object of our study. This system is the basic mechanism of ship maneuvering (acceleration and braking). A failure of the propulsion unit may cause a dangerous event, like a collision with another ship, drifting to shallows, or financial or environmental losses. Such circumstances imply the necessity of monitoring the system and taking appropriate steps in the case of operational faults.

In such systems, apart from fault detection, it is necessary to accomplish the isolation of faults. In particular, fault isolation can refer to a shaft speed sensor or the diesel engine itself. Each of the faults requires different steps to be taken.

A linearized continuous-time model of the ship propulsion system is shown in Fig. 12, where five blocks are distinguished: (a) the propeller-pitch control system, (b) the diesel engine, (c) the shaft, (d) the propeller and ship dynamics, and (e) the PI controller. The description of system parameters is given in Table 3.

The system can be described in the continuous-time domain by the state-space model (17)–(18) with the following matrices:

$$\boldsymbol{A} = \begin{bmatrix} -k_t & 0 & 0 & 0 \\ \dfrac{b_\theta}{M} & \dfrac{b_n}{M} & \dfrac{b_\nu}{M} & \dfrac{1}{M} \\ \dfrac{a_\theta}{m} & \dfrac{a_n}{m} & \dfrac{a_\nu}{m} & 0 \\ 0 & 0 & 0 & -\dfrac{1}{\tau_C} \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} k_t & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \dfrac{k_y}{\tau_C} \end{bmatrix},$$

$$\boldsymbol{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \qquad \boldsymbol{N} = \begin{bmatrix} 0 & 0 \\ -\dfrac{1}{M} & 0 \\ 0 & -\dfrac{1}{m} \\ 0 & 0 \end{bmatrix},$$

$$\boldsymbol{F}_1 = \begin{bmatrix} -k_t & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad \boldsymbol{F}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix},$$

and the following vector signals:

$$\boldsymbol{x} = \begin{bmatrix} \theta \\ n \\ v \\ Q_{\mathrm{eng}} \end{bmatrix}, \ \boldsymbol{u} = \begin{bmatrix} \theta_{\mathrm{ref}} \\ Y \end{bmatrix}, \ \boldsymbol{d} = \begin{bmatrix} Q_f \\ T_{\mathrm{ext}} \end{bmatrix}, \ \boldsymbol{f} = \begin{bmatrix} \Delta\theta \\ \Delta\dot{\theta} \\ \Delta n \end{bmatrix},$$

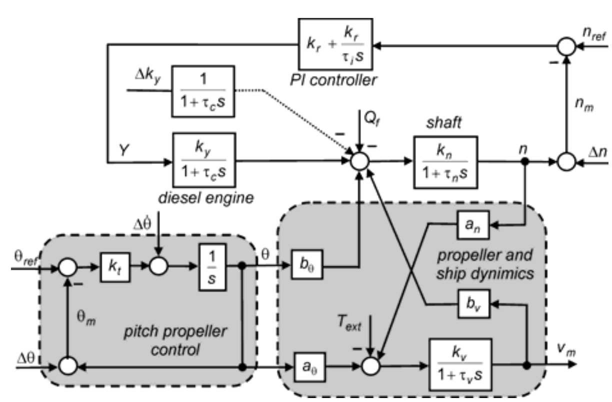


Fig. 12. Linearized model of the ship propulsion system.

$$\boldsymbol{y} = \begin{bmatrix} \theta_m \\ n_m \\ v_m \end{bmatrix}, \quad \tilde{\omega} = \begin{bmatrix} -k_t\omega_\theta \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \omega = \begin{bmatrix} \omega_\theta \\ \omega_n \\ \omega_v \end{bmatrix},$$

where $\boldsymbol{x}$ is the state vector, $\boldsymbol{u}$ denotes the control vector, $\boldsymbol{f}$ stands for an additive fault vector, $\boldsymbol{d}$ denotes an unknown disturbance vector, $\boldsymbol{y}$ is the measurement vector, while the signals $\boldsymbol{w}$ and $\boldsymbol{v}$ have noisy characteristics. The descriptions of θ, n, ν, Q_{eng} and other elements of model vectors and matrices are given in Table 3.

Table 3. Parameters of the diagnosed ship propulsion system.

Symbol	Description
$\theta,\ \theta_m$	Propeller pitch angle and its measurement
θ_{ref}	Set-point for the propeller pitch angle
$\delta\theta$	Pitch-angle measurement fault
$\delta\dot{\theta}$	Leakage
$\nu,\ \nu_m$	Ship speed and its measurement
$n,\ n_m$	Shaft speed and its measurement
δn	Angular-velocity measurement fault
Y	Fuel index (level)
Q_f	Friction torque
Q_{eng}	Torque developed by the diesel engine
T_{ext}	External force representing the influence of wind and waves
$\nu_\theta, \nu_n, \nu_\nu$	Measurement noises for θ, n, ν
M	Shaft inertia
m	Ship weight
K_T	Pitch-angle control gain
k_y	Engine gain
τ_c	Engine time-constant
$a_\theta,\ a_n,\ a_v$ $b_\theta,\ b_n,\ b_v$	Parameters of the steady state

The faults in the examined object are associated with the sensor of the pitch angle of the propeller, the sensor of the angular velocity of the shaft, and the diesel engine itself (Izadi and Blanke, 1998).

The sensor of the pitch angle θ of the propeller can indicate a fault in two cases, i.e., when:

(a) generating too low a signal (a negative deviation $\Delta\theta_{\mathrm{low}}$) due to a broken wire or shaft stack,

(b) generating too high a signal (with a positive deviation $\Delta\theta_{\mathrm{high}}$) due to a broken wire or shaft stack.

Another fault is a hydraulic leakage, which can bring about a slow change of the propeller pitch angle ($\Delta\dot{\theta}$).

The tachometer can generate the following faults:

(a) generating a maximum signal value ($\Delta n_{\max}$) due to an electromagnetic disturbance,

(b) generating a minimum signal value ($\Delta n_{\min}$) due to a signal fade-out in the converter.

The appearance of the above faults can have serious consequences. The fault $\Delta\theta_{\mathrm{high}}$ can decrease the ship velocity, which brings about the risk of maneuvering, and extra operational costs. The failure $\Delta\theta_{\mathrm{low}}$ increases the ship velocity and the danger of a collision. The fault $\Delta n_{\max}$ can impel a decrease in the ship velocity, which results in delayed ship operation and increased operational costs. The fault $\Delta n_{\min}$ gives rise to unplanned acceleration, threatening with a collision.

5.2. Results of Evolutionary Explorations

The vector of the sought real eigenvalues of the matrix $(\boldsymbol{A} - \boldsymbol{KC})$, for $i = 1, 2, \ldots, N$, is expressed as

$$\lambda(\boldsymbol{v}_i) = \boldsymbol{v}_i = \begin{bmatrix} v_{1_i} & v_{2_i} & v_{3_i} & v_{4_i} \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^4,$$

where v_{k_i} is the k-th coordinate of the parameter vector $\boldsymbol{v}_i$, which stands for the i-th individual. The searched ranges of the optimized parameters v_k were set as $v_{1,2_i} \in [-30, -0.5]$ and $v_{3,4_i} \in [-100, -31]$, respectively. For simplicity of implementation and interpretation, at this stage of the study we have applied the evolutionary approach founded on floating-point computer implementation of numbers. As has been done before, the 10% elitism strategy was employed in the EC algorithm in order to sustain effective solutions.

The weighting functions have been taken as:

$$\boldsymbol{W}_1(s) = \operatorname{diag}\left\{\frac{1}{(0.01s+1)(0.05s+1)}\right\},$$

$$\boldsymbol{W}_2(s) = \operatorname{diag}\{1\},$$

$$\boldsymbol{W}_{3,4}(s) = \operatorname{diag}\left\{\frac{(0.01s+1)(0.05s+1)}{(0.001s+1)^2(0.00001s+1)^2}\right\},$$

which allows separating the influence of the faults and the noises. In order to maximize the fault effects at low frequencies and to minimize the noise results in the high frequency range, the matrices $\boldsymbol{W}_1(s)$ have been set as low-pass filters and the functions $\boldsymbol{W}_3(s)$ and $\boldsymbol{W}_4(s)$ composed as suitable high-pass filters.

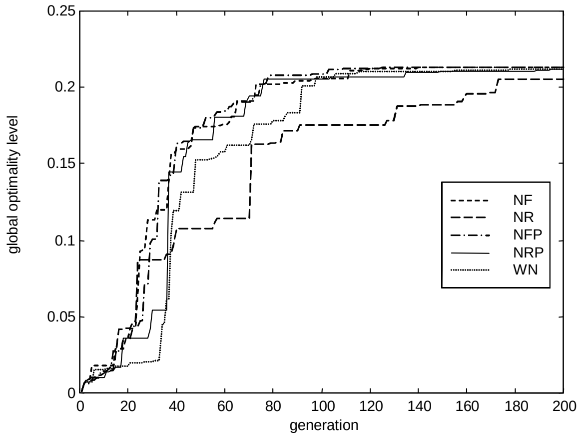


Fig. 13. Highest global optimality level (GOL) obtained by ECs with continuous niching.

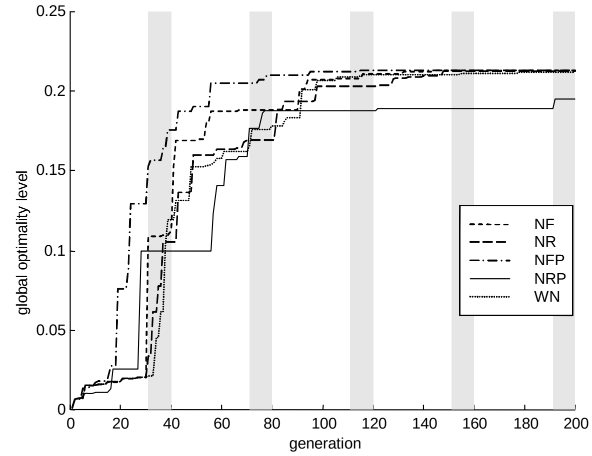


Fig. 14. Best global optimality level with periodic niching.

In the process of evolutionary searching, the four niching methods, NF, NR, NFP and NRP, as well as the algorithm without niching (WN) were considered.

Computations were carried out on the assumption that the ranges of the sought parameters are divided into three parts $(\epsilon = 3)$. With similar values of the other EC parameters, the four-point crossover mechanism was implemented with the probability 0.8. The initial population was placed in a constrained region of the searched space.

Figure 13 portrays some results of optimization performed with the use of the continuous niching mechanism, and, in particular, the evolution of the highest global optimality levels characterizing EC products in consecutive generations.

The effectiveness of the solutions obtained in the case of EC-optimization using the niching mechanisms only periodically switched on is characterized in Fig. 14. The switching took place every 30-th generation for 10 EC cycles as illustrated by the gray stripes in the graphs, which represent the trajectories of the highest global optimality level.

As can be seen from Figs. 13 and 14, the niching applied to the fitness of the members of both the whole population and the parental pool solely (denoted as NF and NFP, respectively) achieve superior outcomes in terms of the highest global optimality levels amongst the solutions. The approaches consisting in niching the individuals' ranks (NR and NRP) lead to weaker convergence, which can be attributed to an undue perturbation of the optimal selections.

Moreover, as results from Figs. 13 and 14, the application of the periodic niching mechanism to the evolutionary search does not influence significantly the achieved highest global optimality level. At the same time, it is clear that the periodic niching mechanism is less time consuming. It is also obvious that very soon, even in the initial generations, the classical EC mechanism without niching (WN) can bring about fast convergence to sub-optimal solutions.

5.3. Statistical Results of the Genetic Search

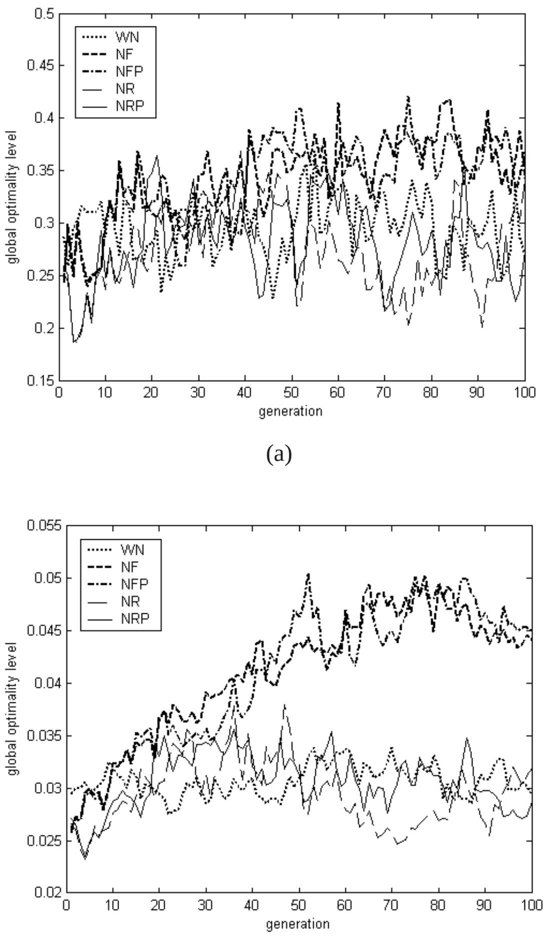
In order to obtain a clear distinction between the results of different niching approaches, another comprehensive statistical investigation considering genetic algorithms has been undertaken. This time, plain GAs without the elitism strategy were implemented with the use of a binary 113-bits genotype of the sought four real eigenvalues (individually represented on 27, 29, 27, and 30 bits, respectively), which set a common accuracy limit to $1e-7$.

The searched ranges of the optimized parameters were set as $v_{1_I} \in [-10\ -0.02]$, $V_{2_i} \in [-100\ -53]$, $V_{3_i} \in [-10\ -0.2]$, $V_{4_i} \in [-100\ -12]$, respectively. All optimization runs started with the same initial population.

Figure 15 shows mean run-time results of GA optimization performed twelve times with continuous niching. Figure 15(a) portrays the evolution of the mean highest global optimality levels of the individuals obtained in consecutive GA generations, whereas the mean value of GOLs computed from averaged GOLs of all the Pareto optimal solutions within each niching version of the GA at a running step is presented in Fig. 15(b).

The same series of experiments were performed for GAs using periodic niching, and the corresponding results are shown in Fig. 16.

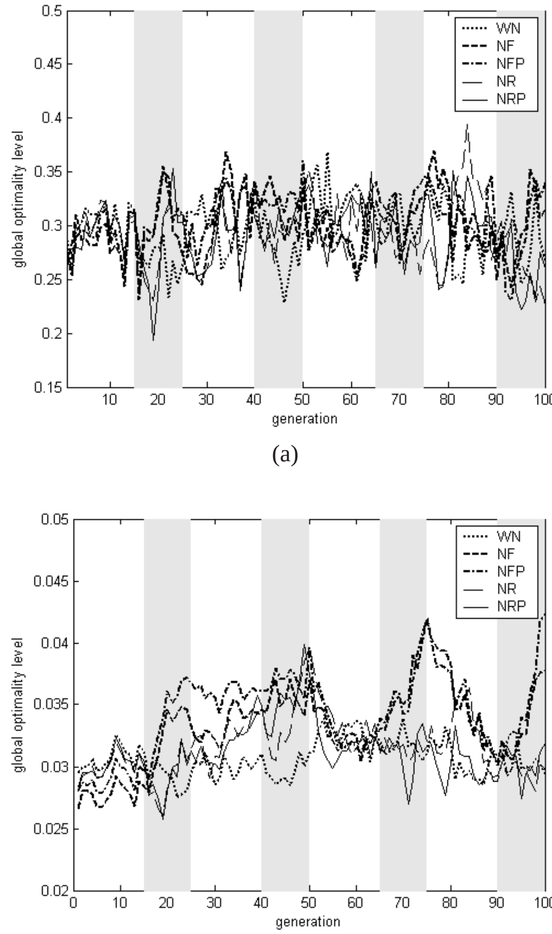
Figures 15 and 16 approve the previously posed supposition that fitness-related niching (NF and NFP) leads to superior outcomes in terms of global optimality. Especially, the improvement is manifested by the plot of the average Pareto optimal GOL. Quite surprisingly, the periodic mechanism of bringing the niching into play gives the clearest substantiation of the sense of niching. As can



(a)

(b)

Fig. 15. Mean GOL optimality (for 12 GA runs with continuous niching): (a) highest and (b) average P-optimal values.



(a)

(b)

Fig. 16. Mean GOL optimality (for 12 GA runs with periodic niching): (a) highest and (b) average P-optimal values.

be observed in Fig. 16(b), each period of improving optimality (coinciding with niching) is followed by a period of loosing it apparently (being a consequence of switching off both the elitism and the niching).

Another way of looking into the problems of the Pareto optimality and diversity is presented in Fig. 17, where a plot of the number of Pareto fronts in GA generations computed as a mean value from the twelve EMO computation runs for each method. Note that the diversification of solutions with respect to the (Pareto) optimality, measured by the number of P-fronts, allows discriminating a smaller set of winners, which is usually a cumbersome issue in the case of high dimensioning of objective spaces (Kowalczuk and Białaszewski, 2001; 2002; 2003).

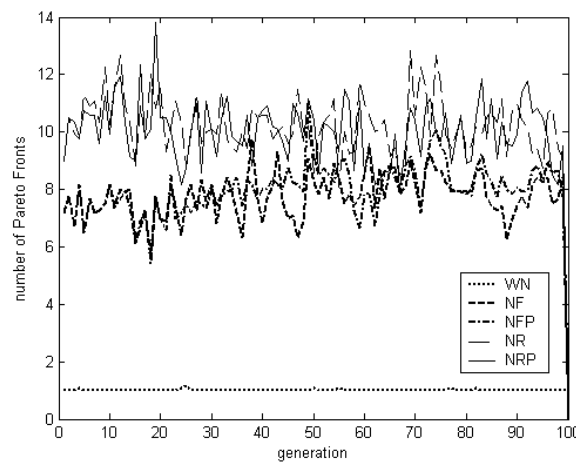


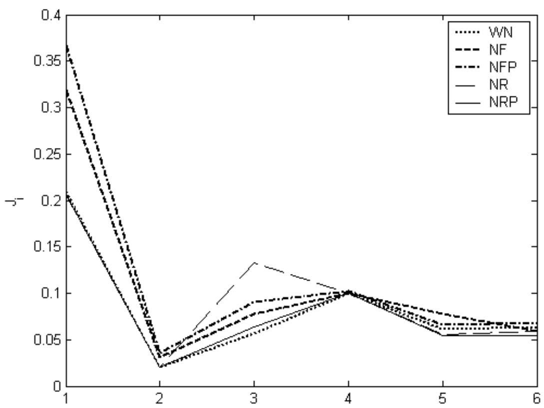
Fig. 17. Mean number of Pareto fronts in GA generations.

An exemplary set of the eigenvalues of the diagnostic observer obtained by the genetic algorithm using continuous niching of fitness functions (NF) as well as its corresponding Pareto optimal index (40) are as follows:

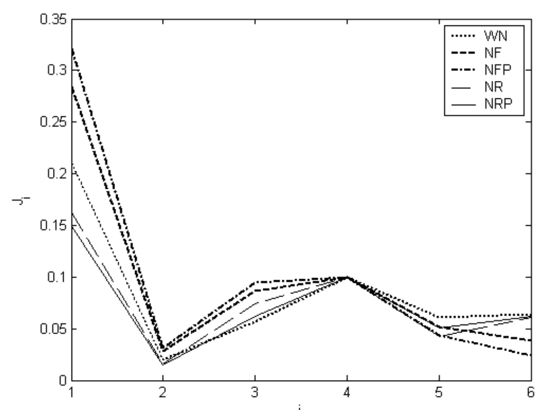
$$\lambda_{\mathrm{NF}} = \big[-9.0904 \ -87.0807 \ -0.4784 \ -41.9664 \big]$$

$$J_{\rm NF} = \big[1.4804e+2\ 2.3243e-6\ 5.8108e-1$$
$$1.0015e+0\ 1.0755e+9\ 8.4750e+7\big].$$

Quite an instructive and practical analysis of the obtained results is offered by Fig. 18, which illustrates the final effect of optimization gained in terms of the searched six-dimensional space. Specifically, each profile represents a set of mean normalized values of the partial-objective functions acquired by each niching variant (using the continuous and periodic timing) of optimizing GA algorithms run twelve times. In other words, consecutive abscissa points correspond to the co-ordinates (32)–(37) of the optimized vector (40) normalized for the purpose of scaling the results (by using the following set of scaling coefficients: [620 1e-4 10 10 1e+10 1e+9]). Note that the first objective has been maximized, while the others have been minimized.



(a)



(b)

Fig. 18. Mean normalized Pareto optimal values of objectives (i) gained in GAs with: (a) continuous; (b) periodic niching.

5.4. Simulated Verification of Optimization Results

Validating simulations of the analyzed diagnostic system were performed with the observer characterized by the obtained solutions of the five GA algorithms. A sequence of additive faults (Izadi and Blanke, 1998; Kowalczuk and Białaszewski, 2004a) in simplified forms is depicted in Fig. 19.

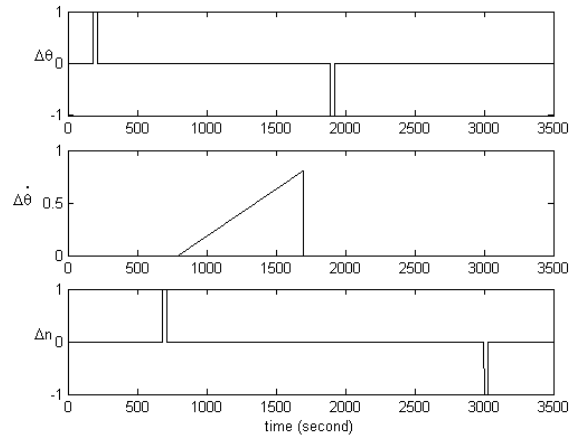


Fig. 19. Additive faults to be detected.

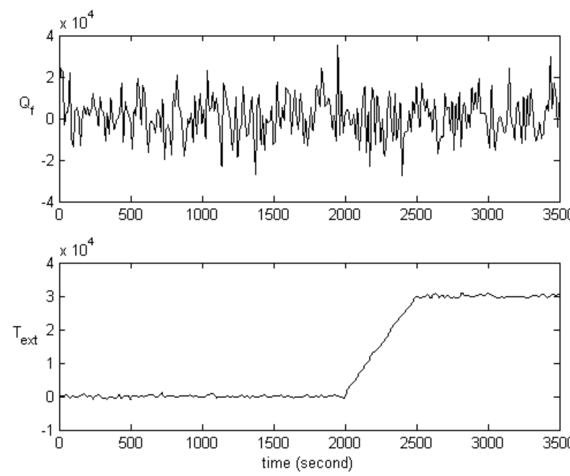
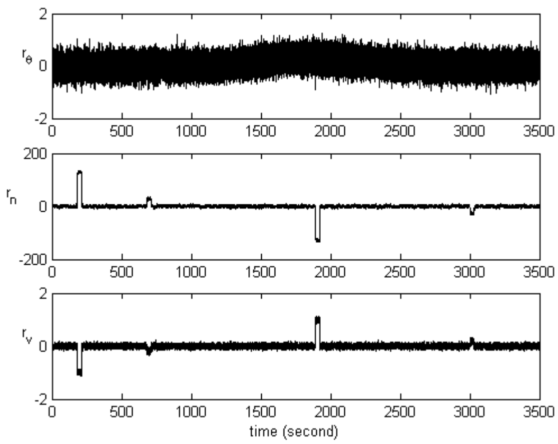


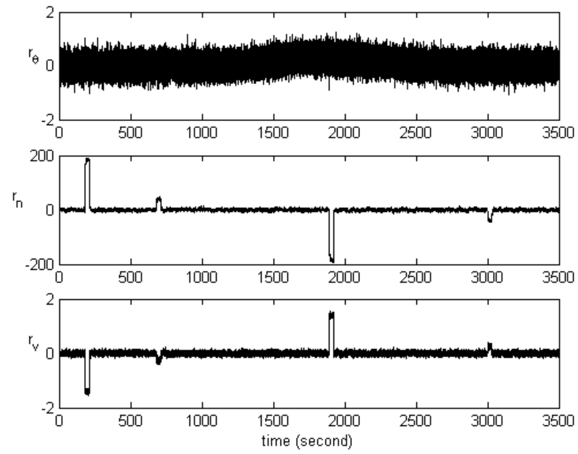
Fig. 20. External disturbances to the process.

The disturbances representing a friction torque and an external force influencing the object are presented in Fig. 20. The noisy signal $\omega(t)$, affecting the states and measurements, was generated as a zero-mean Gaussian white-noise process.

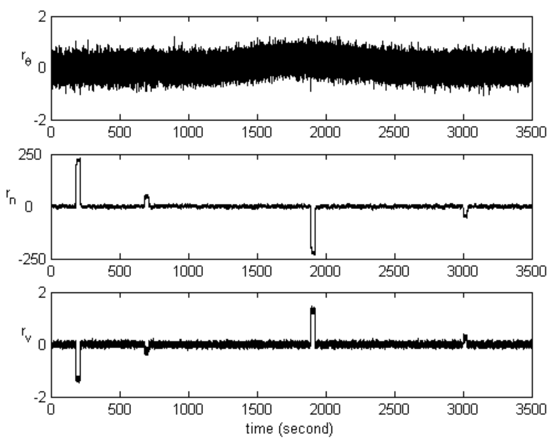
Each of the twelve solutions yielded by each niching GA variant were exercised, and the respective residual signals were used to form a mean behavior of the fault detection observer. These mean residuals $r_\theta = \theta - \hat{\theta}$,



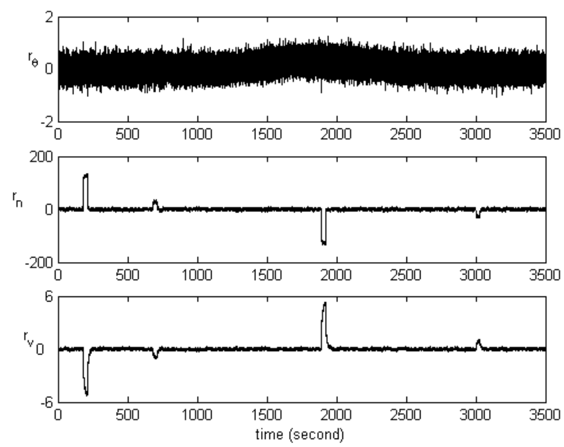
(a)



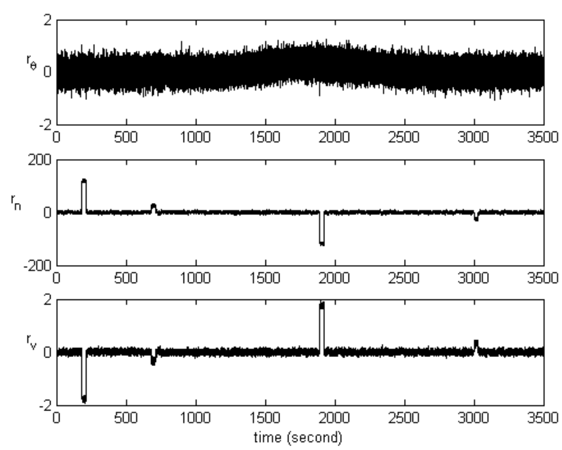
(b)



(c)



(d)



(e)

Fig. 21. Residual signals for the observer obtained with the analyzed methods: WN (a); NF (b); NFP (c); NR (d); NRP (e).

$r_n = n - \hat{n}$, and $r_v = v - \hat{v}$, obtained for particular niching methods, are plotted in Fig. 21. As can be seen from these figures, practically all the faults of Fig. 19 give characteristic symptoms in at least one of the residues. Nevertheless, the solution achieved by the standard GA without niching (WN) appears to be most inferior in terms of the discussed FDI system design.

The residuals demonstrate changes analogous to the generic fault signal applied. It is thus apparent that with the use of appropriate optimal filtration, the symptoms information included in the residuals makes it possible to detect and isolate all the faults.

6. Conclusions

The evolutionary computation (GA/EC) approach proves to be an effective method of solving multi-objective optimization problems, which can be successfully used in the synthesis of engineering systems. It is important that this approach, permitting a global multi-objective search in multi-dimensional spaces for Pareto optimal solutions, is immune to the possible discontinuity or multi-modality of partial objective functions. At the same time, the multiple solutions yielded by the Pareto optimal approach, which is generally criticized for its non-uniqueness, find their full application in the process of ranking and selecting the individuals to parental pools at the end of each evolutionary cycle.

For the purpose of making the final evaluation of the obtained outcomes, the solutions can be ordered with the use of a global optimality index, which gives a scalar measure of each solution relative to the attainable maximum values of all the partial quality indices.

The classical GA/EC mechanism without niching (WN) can bring about fast (often premature) convergence to sub-optimal solutions in the initial generations. Therefore the ranking procedure, supporting the Pareto optimal selection of the best solutions, has been enriched with a niching mechanism, which – as has been shown – can be implemented in various ways taking into account both the objects (the fitness, the rank, the population, the parents) and the timing type (continuous, periodic) of niching. It facilitates more effective exploration of the parameter space, prevents genetic algorithms from their premature convergence, and takes care of solution diversity.

Note that the results of a suitably implemented mechanism of niching also have a simple robustness interpretation, connected with the preservation of niches of densely populated species, despite following the evolutionary policy of ‘uniform breeding’.

As compared to the fitness-based niching, the niching of ranks can be characterized as a ‘stronger’ mechanism. This effect directly results from the change of both the ranks and the domination structure. At the same time, the niching of fitness constitutes solely a source of an indirect modification of the ranks, and changing the fitness vector need not alter its level of domination.

With the niching of the ranks (NRP) and fitness (NFP) of the parental pool, the genetic algorithm is able to find those solutions which are totally Pareto optimal. This effect can be related to the fact that the diversity of evolving generations can then be effectively sustained.

In general, however, niching applied to the fitness of the members of the whole population and the parental pool (NF and NFP) leads to superior results in terms of the global optimality level while, depending on the parameters applied, the niching of the individuals’ ranks (NR and NRP) can lead to weaker convergence to P-optimal solutions, as an effect of undue perturbations in genetic selection.

Clearly, these effects are weakened when using the time-saving periodic niching mechanism in the genetic search. Nevertheless, it may be enough to temporarily invoke such fitness-related niching in order to influence the ability of the GA to achieve higher global optimality.

If the criteria considered are not complex enough, the niching mechanism can easily appear ineffective.

A completely different approach (GGA) to multi-objective optimization problems was considered in (Kowalczuk and Białaszewski, 2001; 2002; 2003).

References

Brogan W.L. (1991): *Modern Control Theory*. — Englewood Cliffs, NJ: Prentice Hall.

Chen J., Patton R.J. and Liu G. (1996): *Optimal residual design for fault diagnosis using multi-objective optimization and genetic algorithms*. — Int. J. Syst. Sci., Vol. 27, No. 6, pp. 567–576.

Chambers L. (Ed.) (1995): *Practical Handbook of Genetic Algorithms*. — Boca Raton, FL: CRC Press.

Coello C.C.A. (2001): *A short tutorial on evolutionary multiobjective optimization*. — Proc. 1st Int. Conf. *Evolutionary Multi-Criterion Optimization*, Lecture Notes in Computer Science, No. 1993, pp. 21-40, Berlin: Springer.

Cotta C. and Schaefer R. (Eds.) (2004): *Evolutionary Computation*. — Int. J. Appl. Math. Comput. Sci., Vol. 14, No. 3, pp. 279–440.

Deb K., Pratap A., Argarwal S. and Meyarivan T. (2000): *A fast and elitist multi-objective genetic algorithm: NSGA-II*. — Techn. Rep., No. 200001 (PIN 208 016), Kanpur, India: Kanpur Genetic Algorithms Laboratory.

De Jong K.A. (1975): *An analysis of the behavior of a class of genetic adaptive systems*. — Ph.D. thesis., Ann Arbor, MI: University of Michigan.

Dridi M. and Kacem I. (2004): *A hybrid approach for scheduling transportation networks*. — Int. J. Appl. Math. Comput. Sci., Vol. 14, No. 3, pp. 397–409.

Fogarty T.C. and Bull L. (1995): *Optimizing individual control rules and multiple communicating rule-based control systems with parallel distributed genetic algorithms*. — IEE Proc. Contr. Theory Applic., Vol. 142, No. 3, pp. 211–215.

Fonseca C.M. and Fleming P.J. (1993): *Genetic algorithms for multi-objective optimization: Formulation, discussion and modification*. — In: (Forrest, 1993), pp. 416–423.

Forrest S. (Ed.) (1993): *Genetic Algorithms*. — Proc. 5th Int. Conf., San Mateo, CA: Morgan Kaufmann.

Goldberg D.E. (1986): *The genetic algorithm approach: Why, how, and what next*. In: Adaptive and Learning Systems. Theory and Applications (K.S. Narendra, Ed.). — New York: Plenum Press, pp. 247–253.

Goldberg D.E. (1989): *Genetic Algorithms in Search, Optimization and Machine Learning*. — Reading, MA: Addison-Wesley.

Goldberg D.E. (1990): *Real-coded genetic algorithms, virtual alphabets, and blocking*. — Techn. Rep., No. 90001, Champaing, IL: University of Illinois at Urbana.

Grefenstette J.J. (Ed.) (1985): *Genetic Algorithms and their Applications*. — Proc. Int. Conf., Pittsburgh, PA: Lawrence Erlbaum Associates.

Hajela P. and Lin C.-Y. (1992): *Genetic search strategies in multicriterion optimal design*. — Struct. Optim., Vol. 4, No. 2, pp. 99–107.

Holland H. (1975): *Adaptation in Natural and Artificial Systems*. — Ann Arbor, MI: University of Michigan Press.

Horn J. and Nafpliotis N. (1993): *Multiobjective optimization using the niched Pareto genetic algorithm*. — Techn. Rep., No. 93005, Genetic Algorithms Laboratory, University of Illinois at Urbana.

Horn J., Nafpliotis N. and Goldberg D.E. (1994): *A niched Pareto genetic algorithm for multiobjective optimization*. — Proc. 1st IEEE Conf. *Evolutionary Computation*, IEEE World Congress *Computational Computation*, Piscataway, NJ, Vol. 1, pp. 82–87.

Huang Y. and Wang S. (1997): *The identification of fuzzy grey prediction systems by genetic algorithms*. — Int. J. Syst. Sci., Vol. 28, No. 1, pp. 15–24.

Izadi-Zamanabadi R. and Blanke M. (1998): *A ship propulsion system model for fault-tolerant control*. — Techn. Rep., No. 4262, Aalborg University, Denmark.

Kirstinsson K. (1992): *System identification and control using genetic algorithms*. — IEEE Trans. Syst. Man Cybern., Vol. 22, No. 5, pp. 1033–1046.

Korbicz J., Kościelny J.M., Kowalczuk Z. and Cholewa W., (Eds.) (2004): *Fault Diagnosis. Models, Artificial Intelligence, Applications*. — Berlin: Springer.

Kowalczuk Z. and Białaszewski T. (2000a): *Pareto-optimal observers for ship propulsion systems by evolutionary algorithms*. — Proc. IFAC Symp. *Safeprocess*, Budapest, Hungary, Vol. 2, pp. 914–919.

Kowalczuk Z. and Białaszewski T. (2000b): *Fitness and ranking individuals warped by niching mechanism*. — Proc. Polish-German Symp. *Science, Research, Education*, Zielona Góra, Poland, pp. 97–102.

Kowalczuk Z. and Białaszewski T. (2001): *Evolutionary multi-objective optimization with genetic sex recognition*. — Proc. 7th IEEE Int. Conf. *Methods and Models in Automation and Robotics*, Międzyzdroje, Poland, Vol. 1, pp. 143–148.

Kowalczuk Z. and Białaszewski T. (2002): *Performance and robustness design of control systems via genetic gender multi-objective optimization*. — Proc. 15th IFAC *World Congress*, Barcelona, Spain, (CD-ROM, 2a).

Kowalczuk Z. and Białaszewski T. (2003): *Multi-gender genetic optimization of diagnostic observers*. — Proc. IFAC Workshop *Control Applications of Optimization*, Visegrad, Hungary, pp. 15–20.

Kowalczuk Z. and Białaszewski T. (2004a): *Genetic algorithms in multi-objective optimization of detection observers*. — In: (Korbicz *et al.*, 2004), pp. 511–556.

Kowalczuk Z. and Białaszewski T. (2004b): *Periodic and continuous niching in genetic optimization of detection observers*. — Proc. 10-th IEEE Int. Conf. *Methods and Models in Automation and Robotics*, Międzyzdroje, Poland, Vol. 1, pp. 781–786.

Kowalczuk Z., Suchomski P. and Białaszewski T. (1999a): *Evolutionary multi-objective Pareto optimization of diagnostic state observers*. — Int. J. Appl. Math. Comput. Sci., Vol. 9, No. 3, pp. 689–709.

Kowalczuk Z., Suchomski P. and Białaszewski T. (1999b): *Genetic multi-objective Pareto optimization of state observers for FDI*. — Proc. Europ. *Contr. Conf.*, Karlsruhe, Germany, (CD-ROM, CP-15:10).

Kowalczuk Z. and Suchomski P. (2004a): *Control theory methods in diagnostic system design*. In: (Korbicz *et al.*, 2004), pp. 155–218.

Kowalczuk Z. and Suchomski P. (2004b): *Optimal detection observers based on eigenstructure assignment*. In: (Korbicz *et al.*, 2004), pp. 219–259.

Li C.J., Tzeng T. and Jeon Y.C. (1997): *A learning controller based on nonlinear ARX inverse model identified by genetic algorithm*. — Int. J. Syst. Sci., Vol. 28, No. 8, pp. 847–855.

Linkens D.A. and Nyongensa H.O. (1995): *Genetic algorithms for fuzzy control, Part 1: Offline system development and application*. — IEE Proc. Contr. Theory Applic., Vol. 142, No. 3, pp. 161–185.

Man K.S., Tang K.S., Kwong S. and Lang W.A.H. (1997): *Genetic Algorithms for Control and Signal Processing*. — London: Springer.

Martinez M., Senent J. and Blacso X. (1996): *A comparative study of classical vs. genetic algorithm optimization applied in GPC controller*. — Proc. IFAC 13th *Triennial Word Congress*, San Francisco, CA, pp. 327–332.

Michalewicz Z. (1996): *Genetic Algorithms + Data Structures = Evolution Programs*. — Berlin: Springer.

Obuchowicz A. and Prętki P. (2004): *Phenotypic evolution with mutation based on symmetric α-stable distributions*. — Int. J. Appl. Math. Comput. Sci., Vol. 14, No. 3, pp. 289–316.

Park D. and Kandel A. (1994): *Genetic-based new fuzzy reasoning models with application to fuzzy control*. — IEEE Trans. Syst., Man Cybern., Vol. 24, No. 1, pp. 39–47.

Patton R.J., Frank P.M. and Clark R.N., (Eds.) (1989): *Fault Diagnosis in Dynamic Systems. Theory and Application*. — New York: Prentice Hall.

Ryan C. (1995): *Niche and species formation in genetic algorithms*, In: (Chambers, 1995). — Vol. 1, No. 2, pp. 55–74.

Schaffer J.D. (1985): *Multiple objective optimization with vector evaluated genetic algorithms*. In: (Grefenstette, 1985). — pp. 93–100.

Silverman B.W. (1986): *Density Estimation for Statistics and Data Analysis*. — London: Chapman and Hall.

Srinivas N. and Deb K. (1994): *Multiobjective optimization using nondominated sorting in genetic algorithms*. — Evolut. Comput., Vol. 2, No. 3, pp. 221–248.

Suchomski P. and Kowalczuk Z. (2004): *Robust H^{∞}-optimal synthesis of FDI systems*, In: (Korbicz *et al.*, 2004), pp. 261–298.

Viennet R., Fontiex C. and Marc I. (1996): *Multicriteria optimisation using a genetic algorithm for determining a Pareto set*. — Int. J. Syst. Sci., Vol. 27, No. 2, pp. 255–260.

Zakian V. and Al-Naib U. (1973): *Design of dynamical and control systems by the method of inequalities*. — IEE Proc. Contr. Theory Applic., Vol. 120, No. 11, pp. 1421–1427.

Zitzler E. and Thiele L. (1998): *An evolutionary algorithm for multiobjective optimization: The Strength Pareto Evolutionary Algorithm*. — Techn. Rep., No. 43, Zurich, Switzerland: Computer Engineering and Networks Laboratory, ETH.

Zitzler E., Laumanns M. and Thiele L. (2001): *SPEA-2: Improving the strength Pareto evolutionary algorithm*. — Techn. Rep., No. 103, Zurich, Switzerland: Computer Engineering and Networks Laboratory, Dept. of Electrical Engineering, ETH.

Received: 18 October 2005
Revised: 8 January 2006