THE JOURNAL OF
COGNITIVE SYSTEMS
VOLUME 01 NUMBER 01 DECEMBER 2016

PRECISION EVOLUTIONARY OPTIMIZATION PART I: NONLINEAR RANKING APPROACH

Özer Ciftcioglu[1] , S. Serhat Seker[2], Jelena Dikun[3], Emine Ayaz[2]

[1] Delft University of Technology, Delft, The Netherlands
[2] Istanbul Technical University, Istanbul, Turkey
[3] Lithuanian Maritime Academy, Klaipeda, Lithuania

Theoretical foundations of a robust approach for multiobjective optimization by evolutionary algorithms are introduced. The optimization method used is the conventional penalty function approach, which is also known as bi-objective method. The novelty of the method stems from the dynamic variation of the commensurate penalty parameter for each objective treated as constraint. The parameters collectively define the right slope of the tangent as to the optimal front during the search. The slope conforms to the theoretical considerations so that the robust and fast convergence of the search is accomplished throughout the search up to micro level in the range of 10^{-10} or beyond with precision as well as with accuracy thanks to a robust probabilistic distance measure established in this work. The measure is used for nonlinear ranking among the population members of the evolutionary process, and the method is implemented by a computer program called NS-NR developed for this research. The effectiveness of the method is exemplified by a demonstrative computer experiment minimizing a highly non-linear, non-polynomial, non-quadratic etc. function. The algorithm description in detail and further several applications are presented in the second part of this research. The problems used in computer experiments are selected from the existing literature for comparison while the experiments carried out and reported here to demonstrate the simplicity vs effectiveness of the algorithm.

Index Terms— Evolutionary algorithm, multiobjective optimization, constraint optimization, probabilistic modeling.

I. INTRODUCTION

EVOLUTIONARY computation is ubiquitous, due to its effectiveness in many multi-objective optimization problems, spanning all engineering disciplines and the cognitive science. Because of its heuristic nature, and therefore simplicity, it can easily be implemented. Evolutionary computation implies a series of heuristic algorithms which are subject to modification during the search to enhance their effectiveness in a problem solving situation. A very effective heuristic search algorithm known as genetic algorithm (GA) is a special form of evolutionary computation having its search parameters fixed. In this context there are new evolutionary computation methods, which are trying to be competitive with the existing ones, such as differential evolution [1, 2]. Due to the random search mechanism in heuristic optimization algorithms, the exact tracing of convergence of the algorithm to a minimum or maximum, is not possible. However, they are remarkably fast and robust to find a minimum or maximum due to effective search rules embedded in the algorithms. For detailed description for such algorithms mention can be made of some text books [3-5].

Again, because of the heuristic nature of such search algorithms, there are continuous improvements on the heuristics and they are regularly reported in the literature, e.g. [6, 7]. Multi-objective optimization problems may involve plain multi-objectivity, as well as multi-objectivity with constraints. In particular, a single-objective problem with several constraints can be cast into a bi-objective optimization problem. One effective method to deal with single objective and constraints imposed on it is known as penalty function method. In this method the penalty function is simply a function representing the constraint violation, and this function is added to the single objective function that the summation is subjected to minimization. Here there is also a penalty parameter, which determines the appropriate proportion of the violation during the search. The appropriate proportion here is dependent on the progress by the search algorithm, and the nature of the problem. Therefore, the penalty parameter can be considered constant, but in an evolutionary sense it can be adapted during the search. Although the adaptation of the penalty parameter is an appealing concept, an effective method dealing with adaptivity is an issue, and it is subject to investigation in general. [8, 9].

The subject matter of this work is an optimization dealing with a single objective with constraints using the penalty function method, proposing a new effective approach for convergence. In the approach, the random solutions are modelled using probabilistic considerations, to establish a nonlinear distance measure. It is used for effective, i.e. robust ranking of genetic population members and efficient, i.e. fast converging, and stable solutions. The measure is used for nonlinear ranking of the population members during the evolutionary process, and the method is implemented by a

Manuscript received September 5, 2016; accepted December 5, 2016.
Corresponding author: Ozer Ciftcioglu (E-mail: o.ciftcioglu@tudelft.nl).
Digital Object Identifier:

computer program called **NS-NR** (nondominated sorting-nonlinear ranking) algorithm developed for this research.

The research is organized in two parts. In the first part, namely in this work at hand, the theory of the approach is presented with a demonstrative example afterwards. In the second part [10], based on the theoretical considerations, the development of the algorithm is given in detail and some demonstrative optimization problems are presented as applications. The organization of the paper is as follows. In section two, formulation of general multiobjective optimization problem as constraint single objective problem and probabilistic constraint handling are presented. In section three, implementation of the probabilistic constraint handling by means of evolutionary algorithm is given. In section four, the important implications of the probabilistic modeling are highlighted. In section five a demonstrative computer experiment is given and it is followed by discussion and conclusions.

II. MULTIOBJECTIVE OPTIMIZATION BY WEIGHTING METHOD

A. Problem Statement

Weighting method is a known approach for multi-objective optimization problems [11-13]. In this method each objective has an associated weighting coefficient, and the weighted sum of the objectives is minimized. By doing so, the multiple objective functions are rendered to a single objective function. We assume that the weighting coefficients w_i are real numbers such that $0 \leq w_i$ for all objectives i=1,….,k , so that a weighting problem can be stated as

$$\min \sum_{i=1}^{k} w_i f_i(\boldsymbol{x}) \quad \textit{subject to} \quad \boldsymbol{x} \in S \tag{1}$$

Referring to the optimization involved in this work, there is one objective with some constraints. Therefore the problem can be written of the form

$$\min \quad f(\boldsymbol{x}) \;\; \textit{subject to} \;\; g(\boldsymbol{x}) = [g_1(x), g_2(x), ..., g_m(x)]^T \tag{2}$$

The feasible region is assumed to have the form

$$S = \{x \in R^n \mid \; g(\boldsymbol{x}) = [g_1(x), g_2(x), ..., g_m(x)]^T \leq 0\} \tag{3}$$

Considering that, the summation of the constraint violations is as another objective subject to minimization, the problem formulation becomes a problem of two objective functions subject to minimization. The formulation of the problem in this case becomes

$$\min \;\; w_1 f(\boldsymbol{x}) + w_2 G(\boldsymbol{x}) \tag{4}$$

where

$$G(\boldsymbol{x}) = \sum_{i=1}^{k} \mu_i g_i(\boldsymbol{x}) \tag{5}$$

From above we write

$$\begin{gathered} \min \;\; f(\boldsymbol{x}) + \sum_{i=1}^{m} \mu_i g_i(\boldsymbol{x}) = f(\boldsymbol{x}) + G(\boldsymbol{x}) \\ S = \{x \in R^n \mid \; g(\boldsymbol{x}) = [g_1(\boldsymbol{x}), g_2(\boldsymbol{x}), ..., g_m(\boldsymbol{x})]^T \leq 0\} \end{gathered} \tag{6}$$

where w_1=1, w_{2i}=μ_I . In this problem formulation it is clear that the optimization problem turns out to be a constraint optimization with single objective $f(\boldsymbol{x})$, and the constraints denoted by $g_j(\boldsymbol{x})$, where the index j is connected to the associated constraint. This approach is known as *ε-Constraint* method [13, 14]. One of the objective functions is selected to be optimized and all the other objective functions are converted into constraints by setting an upper bound to each of them. Hence, the problem is converted to be of the form minimize $f_l(x)$; subject to $f_j(x) \leq \varepsilon_j$ for all $j=1,2,….,k$, $j \neq l$; $x \in S$ where $l \in \{1,…,k\}$. Naturally, inequalities can be converted to equalities by taking $\varepsilon_j=0$ for all $j=1,2,….,k$, $j \neq l$.

B. Penalty Function Method

Referring to (6) we can write

$$\min \; P(\boldsymbol{x}, R) = f(\boldsymbol{x}) + \sum_{i=1}^{J} R_j g_j(\boldsymbol{x}) \tag{7}$$

where function $g_j(\boldsymbol{x})$ is the *penalty function,* and the parameters R_j are the associated *penalty parameters*, which are not known. If we define a representative penalty parameter, *(R)* representing all the penalty parameters, then (7) turns out to be

$$\min \;\; P(\boldsymbol{x}, R) = f(\boldsymbol{x}) + R \sum_{i=1}^{J} g_j(\boldsymbol{x}) \tag{8}$$

or taking $f_1(\boldsymbol{x}) = f(\boldsymbol{x})$ and the summation of the $g_j(\boldsymbol{x})$ functions as $f_2(\boldsymbol{x})$, (8) becomes

$$P_{opt} = \min\{ f_1(\boldsymbol{x}) + R\, f_2(\boldsymbol{x})\} \tag{9}$$

In order to solve the optimization problem (9) by means of the weighting method, there are some options, as given below.

- *R* is constant. In this case the development of the optimal front is illustrated in figure 1. The optimal point is denoted by P_{opt} subject to obtain by the final development. A solution during the optimization process is denoted by *T* which is far from the P_{opt}. It is to note that *T* is on a Pareto front, and the tangent passing from the point *T* intersects f_2 indicates that indeed, *T* is far from $P_{opt.}$. Seeing

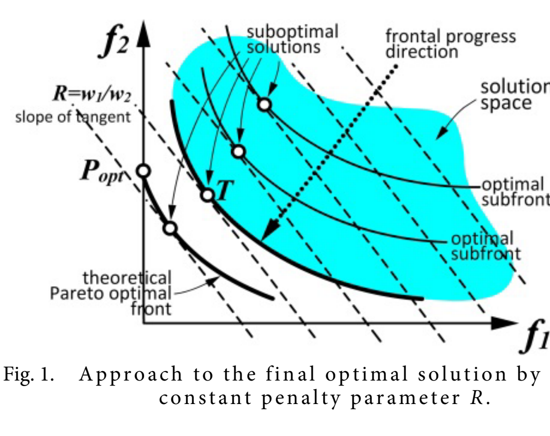


Fig. 1. Approach to the final optimal solution by means of constant penalty parameter *R*.

the problem of convergence to P_{opt}, is a real one, an effective method other than slope of the tangent $R=w_1/w_2$ should be developed. This is because otherwise evolutionary computation needs to be tailed-up by some gradient-based local search algorithm to reach the optimal point. In this case the convergence is essentially due to the constraints and not due to the single objective, leaving the objective in a marginal position with respect to the constraints. Such a case makes the penalty parameter R critical and unpredictable.

+ To determine the penalty parameter with adaptation by means of an extrapolation polynomial. In this case a polynomial is fitted to the optimal front and its extrapolated intersection with the objective function axis is used for the slope of the tangent which is the reasonable estimation of the penalty parameter R. However, in this case, search algorithm tends to move to the straightforward solution, which is the gradual diminishing of the slope as illustrated in figure 2. As result of this option the penalty parameter takes smaller values during the search and may eventually vanish. In the extreme, R goes to zero and problem turns out to be a single objective optimization omitting the constraints.

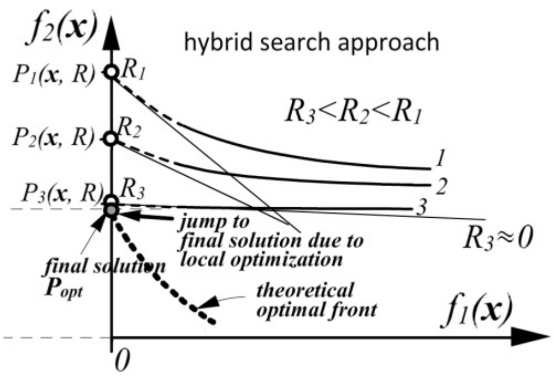


Fig. 2. Approach to the final optimal solution by means of penalty function approach, where R is the penalty parameter being estimated through curve fitting

C. Penalty Parameter

In this subsection, it is aimed to establish the penalty parameter by approximating the Pareto front with respect to $f_1(\boldsymbol{x})$ and $f_1(\boldsymbol{x})$, and to determine the penalty parameter as a slope of a tangent line, the envelope of which is the Pareto front. The parametric representation of the tangent is given by

$$\frac{f_2(\boldsymbol{x})}{t}+\frac{f_1(\boldsymbol{x})}{P_{opt}-t}=1 \tag{10}$$

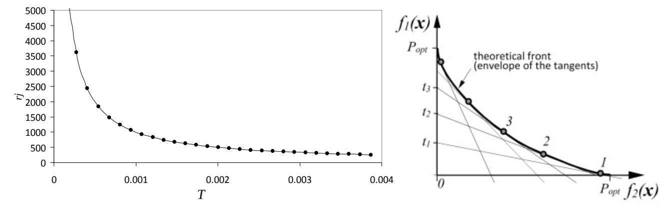
where t is the parameter. In (10), P_{opt} is the optimum solution where $f_2(\boldsymbol{x})= P_{opt}$ and $f_1(\boldsymbol{x})=0$. *From (10),* we write

$$f_2(\boldsymbol{x})=\frac{t}{t-P_{opt}(\boldsymbol{x})}\,f_1(\boldsymbol{x})+t \tag{11}$$

The slope in (11) is given by

$$r=\frac{t}{t-P_{opt}(\boldsymbol{x})} \tag{12}$$

as a *new* penalty parameter, whose variation is shown in figure 3a. The envelope, which approximately represents the Pareto front, is shown in figure 3b.



(a) (b)

Fig. 3. The variation of the *new* penalty parameter $r=(P_{opt}-T)/T$ where $T=P_{opt}-t$ (a); The envelope of tangent and the *new* penalty parameter r (b).

Explicitly, r is the gain in $f_1(\boldsymbol{x})$ per unit decrease in $f_2(\boldsymbol{x})$ at the point of tangent F and within infinitesimally small interval of $f_2(\boldsymbol{x})$. The Pareto front is to obtain by arranging (11) with respect to t and admitting a single solution for it; namely,

$$t^2+[f_1(x)-f_2(x)-P_{opt}(x)]t+f_2(x)P_{opt}(x)=0 \tag{13}$$

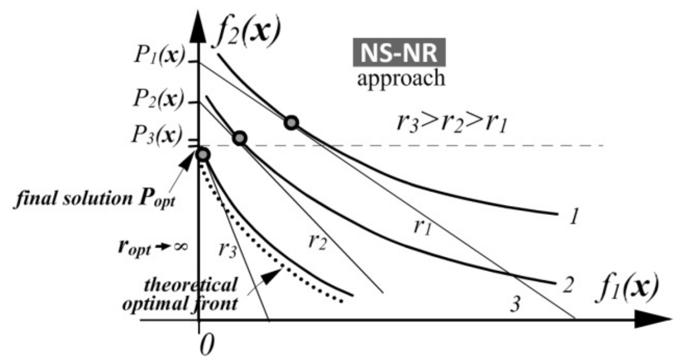


Fig. 4. NS-NR approach to the final optimal solution by means of penalty function approach; r is the penalty parameter.

then, the optimal front is obtained by equating the discriminant to zero that gives the envelope of the tangent as the optimal front.

$$[f_1(x)-f_2(x)-P_{opt}(x)]^2-4f_2(x)P_{opt}(x)=0 \tag{14}$$

The new penalty parameter is zero for $t=0$ and it monotonically increases as t increases. For $t=P_{opt}$ the penalty parameter r goes to infinity. This is sketched in figure 4.

The convergence approach conforming to (12) presents two insights:

+ Approach to optimum is systematic and therefore robust without precarious tangent slope computations
+ No local search for P_{opt} is necessary.

Implementation of the approach is due to a probabilistic modeling of the random solutions in the evolutionary computation and ensuing nonlinear ranking, which are presented in the following section

III. NONLINEAR RANKING BY PROBABILISTIC MODELING

A general constrained optimization problem can be formulated as

$$\min \; P(\boldsymbol{x}) = f(\boldsymbol{x}) + \sum_{i=1}^{J} \mu_j g_j(\boldsymbol{x}) \tag{15}$$

considering (6). Above $f(\boldsymbol{x})$ is the single objective function to be minimized; $g(\boldsymbol{x})$ is the violation of the g_i-th constraint, namely penalty function, μ_i is the associated parameter of the penalty function. At each generation, the evolutionary algorithm tries to make vanish $g_j(\boldsymbol{x})$ during the evolutionary minimization process. Regarding the population density of solutions during the search, the probability density of $g_j(\boldsymbol{x})$ is highest about zero violations, and its value gradually diminishes proportional with the degree of violation. Based on the randomly generated population of the evolutionary algorithm, we can model the violations as a random variable, where the violations are independent due to random population formation by the random composition of chromosomes at each generation. The number of violations per unit violation gradually decreases with the degree of violation conforming to the commensurate number of chromosomes created by the elitism and sorting strategy in the genetic algorithm. This probabilistic pattern continues in the same way without change throughout the generations. The probabilistic description of this process can be modeled by the exponential probability density (pdf), because of its memorylessness property. The property of being the exponential pdf remains the same during the search, being independent of the progress of search process. The exponential pdf is a unique density having this property. Therefore we model the constraint function $g(\boldsymbol{x})$ having an exponential pdf, which is given by

$$f_\lambda(y) = \lambda e^{-\lambda y} \tag{16}$$

where λ is the decay parameter. Denoting

$$y = g_j(x) \tag{17}$$

the pdf in (16) can be written as

$$f_{g_j}(g_j) = \lambda_j e^{-\lambda_j g_j} \tag{18}$$

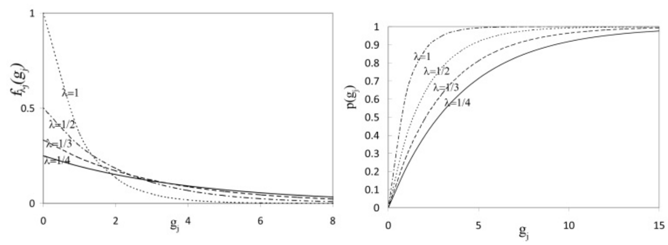
The mean value of the exponential pdf function is equal to λ_j^{-1}. During the evolutionary search $g_i(x)$ is a general form of violation which applies to any member s of the population although s is not explicitly denoted. However, in explicit form, we can write

$$f_{g_j}(g_{j,s}) = \lambda_j e^{-\lambda_j g_{j,s}} \tag{19}$$

where s denotes a population member. We can characterize the exponential pdf function according to the constraint j simply by equating the mean value of the violations g_j to the mean of the exponential pdf, namely

$$\lambda_j = 1/\bar{g}_j \tag{20}$$

One should note that the mean of the exponential probability density of g_j is equivalent to the mean of a uniform probability density applied to the violations g_j. Therefore the mean of the exponential density function is estimated by taking the mean of the violations which are from a uniform probability density and they are independent. Variation of the exponential pdf for different decay parameters is shown in figure 5a.



(a) (b)

Fig. 5. Plot of exponential pdf for different decay constants vs j-th violation g_j (a); $p(g_j)$ vs g_j (b)

Since a violation g_j spans all the violations starting from zero up to the point g_j, the probability of the violation is expressed as cumulative distribution function whose implication is easy to comprehend by considering the extremes. The cumulative distribution function of (16) is given by

$$p(g_j) = \frac{1}{\bar{g}_j} \int_0^{g_j} e^{-\frac{g_j}{\bar{g}_j}} dg_j = 1 - e^{-\frac{g_j}{\bar{g}_j}} \tag{21}$$

The variation of $p(g_j)$ vs g_j with respect to the mean of g_j is shown in figure 5b. For $g_j=0$ violation is zero and for $g_j=\infty$, violation is 1, i.e., 100%. Explicitly $p(g_j)$ is the probability of a violation in the range zero and g_j. It is monotonically increasing function complying with the boundary conditions of $g_j(x)$ which varies between zero and infinity. It is interesting to note that, from the figures, for zero constraint violation the exponential probability density is maximum and probability of violation is minimum.

The probability $p(g_j)$ is an appropriate measure for the magnitude or effectiveness of a violation, and it can be considered as a *probabilistic distance function* or a *metric* measuring the distance from the zero violation fulfilling all the conditions to be a distance measure [15, 16]. Therefore in this work, in (6), μ_j is replaced by $Cr_j(g_j)$ in the form

$$C r_j(g_j) = \mu_j(g_j) \tag{22}$$

So that (21) becomes

$$P(\boldsymbol{x}) = f(\boldsymbol{x}) + C \sum_{i=1}^{J} r_j(g_j) g_j(\boldsymbol{x}) \tag{23}$$

where C is constant common for all the constraints which is called as *convergence parameter* as it is related to the convergence properties of the search; r_j is a *new* penalty parameter which is a function of g_j, in general, and therefore we denote it as

$$r_j = f\,(g_j) \tag{24}$$

In (23), $r_j(g_j)g_j$ is replaced by $p(g_j)$, in the form

$$r_j(g_j)g_j = p_j(g_j) \tag{25}$$

so that (23) becomes

$$P(\boldsymbol{x}) = f(\boldsymbol{x}) + C\sum_{i=1}^{J} p_j(g_j(\boldsymbol{x})) \tag{26}$$

In view of (25), r_j is given by

$$r_j = f(g_j) = p_j(g_j)/g_j \tag{27}$$

The new formulation (26) yields favourable, far reaching implications which are presented below. From (6), we define

$$\sum_{1}^{J}\mu_j g_j = G = C\sum_{1}^{J} p(g_j) \tag{28}$$

where J is the number of constraints; C is a common constant. The probability $p(g_j)$ controls the penalty parameter r_i, which is absorbed in $p(g_j)$ in the form of μ_i. The parameter r_i varies theoretically between zero and infinity, while $p(g_j)$ varies between zero and unity. This nonlinear function transformation $p(g_j)$ plays important role, as it is used for ranking the population members during the genetic search. We can interpret $p(g_j)$ (28) in several ways as follows.

+ On one hand it is a penalty function obtained by a nonlinear interpolation applied to g_j. In this process, the probabilistic considerations apparently are exercised as a nonlinear transformation to the penalty function $g(x_j)$ to obtain another penalty function $p(g_j)$ in order to bring $g(x_j)$ from an infinite range to a finite range namely, between zero and unity.
+ As another interpretation, the penalty function $p(g_j)$ is the probability of a random variable G_j, namely cumulative probability of an exponentially distributed random variable.
+ Yet another interpretation is to consider $p(g_j)$ as another stochastic variable Y_j obtained from a function of stochastic variable X_j.

The last interpretation is highlighted in this work so that several essential implications can be derived. For this aim let us define

$$p(g_j) = H(g_j) \tag{29}$$

where g_j is a random variable. The probability density of this random variable is exponential density function given by (16). According to (29), the new random variable $p(g_j)$ is given by

$$p(g_j) = H(g_j) = \int_0^{g_j} \lambda e^{-\lambda g_j}\, dg_j \tag{30}$$

which gives

$$p(g_j) = H(g_j) = 1 - e^{-\lambda g_j} \tag{31}$$

where $H(g_j)$ is the function of random variable g_j. The probability density $f_p(p)$ of the new random variable p is given by

$$f_p(p) = \frac{f_{g_j}(g_j)}{\left|\dfrac{dH(g_j)}{dg_j}\right|_{g_j = H^{-1}(p)}} \tag{32}$$

that gives

$$f_p(p) = 1 \tag{33}$$

which is a uniform pdf. This surprising result has far reaching implication as this will be seen shortly afterwards, as this is presented in the following section.

IV. IMPORTANT IMPLICATIONS OF THE PROBABILISTIC MODELLING

A. ADAPTIVE ZOOMING FOR RANKING WITH PRECISION

Adaptive zooming for ranking with precision is accomplished by accurate ranking the favourable solutions in the range zero and unity as probabilistic distances, even though the actual constraint values may be close to the optimal point as much as the computer precision can allow, say at the range of 10^{-10}. To illustrate this, a sketch of the Pareto front at the early stage of the genetic search is shown in figure 6a. A sketch of the Pareto front at the last stage of the genetic search is given in figure 6b.

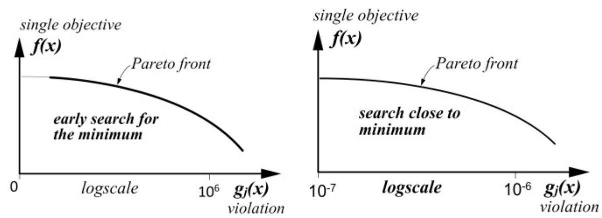


Fig. 6. Sketch of formation of the Pareto front at the early stage (a); at the at the last stage of the GA search (b).

The probabilistic distance to the minimum is illustrated as a typical example in figure 7a by the indicated area where the computation of the gray area is very precarious at the tournament selection process due to the issue of both exact parameterization of the exponential pdf in the existing range and the finite machine precision as well as the finite genotype

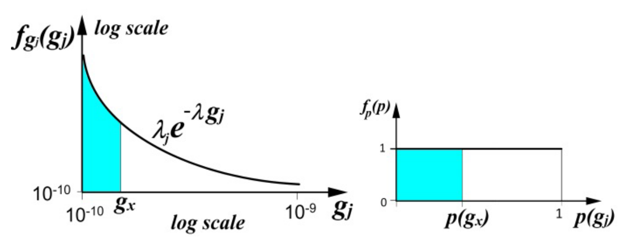


Fig. 7. Mathematical lense; pdf of the violations in the objective functions space (a); in the probabilistic space (b).

coding. This situation is circumvented in figure 7b by taking simply $p(g_j)$ as the probability distance to the minimum. The indicated areas in figure 7a and 7b are the same and they are equal to $p(g_j)$. The grey area in figure 7a, is represented in figure 7b by the probabilistic distance function $p(g_j)$ which varies between zero and unity. This means if the penalty function to be minimized can be close to the optimal point in a micro scale, say in the range of 10^{-10}, the minimization process i.e., tournament selection and ranking of the random solutions takes place in a macro scale in the probabilistic space as shown in figure 7b. This situation is equivalent to apply a commensurate magnifying glass to the space formed by actual objective function and the constraints functions to carry out the convergence process without being affected by any scale of convergence happening in this space. The Pareto front at this micro scale is shown in figure 6b.

B. Effective Tournament Selection

Following the non-dominated sorting procedure as described in [17], an adaptive *threshold of productive chromosomes* is devised both in the non-dominated sorting (NS) stage as well as non-linear ranking (NR) stage of the NS-NR algorithm. The details of the algorithm are given elsewhere [18]. The adaptive threshold of productive chromosomes is based on the sum of the mean of the constraint violations g_T given by

$$g_T = n_{b_j} \sum_{j=1}^{J} \bar{g} = \sum_{j=1}^{J} \frac{n_{b_j}}{\lambda_j} \tag{34}$$

where $n_{bj} = \ln 2/\lambda_j$ which is a constant. Referring to figure 8, the tournament selection, i.e., productive chromosomes selection is accomplished as follows.

a) If the violations of a pair of population members are larger than the threshold, then the solution which has smaller violation wins the competition

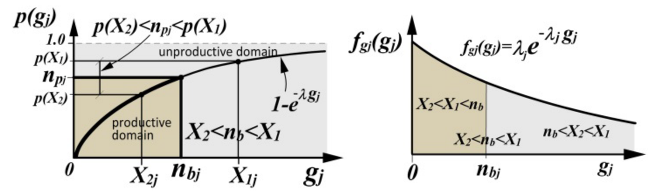
b) If the violations of a pair of population members are smaller than the threshold, then the solution with rank properties in terms of Pareto rank and crowding during the NS stage, or in terms of $P(g_j, \boldsymbol{x})$ rank during NR stage, wins the tournament.

c) If the violations of a pair of population members are at either side of the threshold, then the elite population member that is the chromosome with violation lower than the threshold is selected irrespective to its rank in the NS or NR procedures.

In figure 8 the horizontal axis refers to NS (nondominated sorting) procedures and vertical axis refers to NR (nonlinear ranking) procedures; $n_{bj}=\ln 2/\lambda_j$ is the median of the exponential pdf as shown in figure 8b. For $n_{bj}=\ln 2/\lambda_j$, its counterpart in terms of the probabilistic distance is $n_{pj}=0.5$ which is, in contrast to n_{bj}, a constant. Thus, the constant probabilistic distance measure provides an adaptive threshold for productive chromosomes throughout the generations, at any scale permitted by the machine or genotype precision. By means of this particular tournament selection procedure, the dominance of the average violation by the stiff constraints, that is, by the members with high violations, is prevented; namely, during two consecutive generations the progressive diminishing of the average is aimed against the contingent average increase that may occur especially during the advanced stages of the convergence. In the tournament selection, the domains considered separately are illustrated in figure 8b. The smaller total mean of the constraint violations implies improved convergence to the optimum.

Referring to figure 8b, the probability P_j of the event relevant to the case (c) above is given by

$$P_j = P(g_j) = P(X1_j)P(X2_j) = e^{-\lambda_j n_{bj}} - e^{-2\lambda_j n_{bj}} \tag{35}$$



(a) (b)

Fig. 8. Illustration of the threshold assessment for the tournament selection in both NS and NR procedures.

The variation of P_j with respect to n_{bj} is illustrated in figure 9, in terms of its counterpart p_j which has a maximum at $n_{pj}=0.5$ for $n_{bj}=\ln 2/\lambda_j$. It is to note that, the plot remains the same throughout the generations, although the same plot in the actual violations domain, that is, in the g_j domain corresponds to a family of plots with respect to the parameter λ_j. Implementation of (35) in the NS-NR algorithm is as follows. Should the case (c) arise, the chromosome at the productive domain wins in the tournament selection. The details of this implementation is described in the second part of this sequel [10].

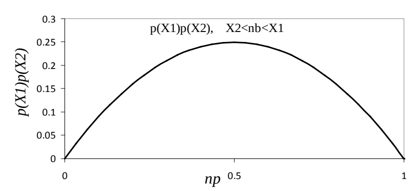


Fig. 9. Plot of the probability that two solutions occur on different sides of the thtreshold n_{bj} vs n_{pj}

C. Fast and robust convergence

With the probabilistic distance providing nonlinear ranking we obtain robust progress for convergence at each generation. To see this, from (27)

$$r_j = \frac{p(g_j)}{g_j} = \frac{1-\mathrm{e}^{-\lambda_j g_j}}{g_j} \tag{36}$$

In the limiting case, i.e., convergence to the minimum, r_j becomes

$$\lim_{g_j \to 0} r_j = \frac{p(g_j)}{g_j} = \lim_{g_j \to 0} \lambda_j e^{-\lambda_j g_j} = \lambda_j \tag{37}$$

The variation of the penalty parameter r_j with g_j, based on (36) is shown in figure 10. In the figure the values of λ_j=10000 and P_{opt} = 1.0. In the same figure, also plot of $r=(P_{opt}-T)/T$ from figure 3a, is also plotted for comparison. The two plots are remarkably almost the same, although their origins of definitions are totally different.

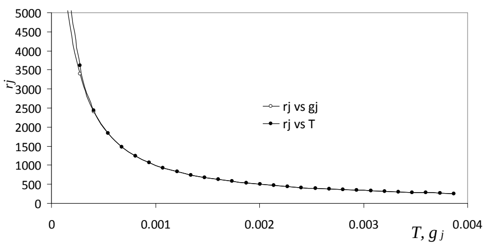


Fig. 10. Illustration of the *new* penalty parameter *r* as to probabilistic modeling: $r=(1-\exp(-\lambda g))/g$ and as to bi-objective formulation: $r=t/(P_{opt}-t)$

V. COMPUTER EXPERIMENT

Computer experiments have been carried out using a standard optimization problem from the literature. The following problem is due to Koziel and Michalewicz [19]. The problem consists of a single objective with two constraints, subject to minimization, as given by (38)-(40).

$$Minimize\ \ f(\boldsymbol{x}) = -\left| \frac{\sum_{i=1}^{n} \cos^4(x_i) - 2\prod_{i=1}^{n} \cos^2(x_i)}{\sqrt{\sum_{i=1}^{n} i x_i^2}} \right| \tag{38}$$

subject to

$$g_1(\boldsymbol{x}) = 0.75 - \prod_{i=1}^{n} x_i \le 0$$
$$g_2(\boldsymbol{x}) = \sum_{i=1}^{n} x_i - 7.5n \le 0 \tag{39}$$

$$where\ \ 0 \le x_i \le 10\ \ (i = 1, \ldots, 20) \tag{40}$$

The best known optimum is
$f(\boldsymbol{x}^*)$=-0.80361910412559 [20], and $f(\boldsymbol{x}^*)$=-0.803553 [19] while Koziel and Michalewicz using Evolutionary Algorithms with the method of homomorphous mappings report their best result as 0.79953 [19]. The variables for the best known solution are given by [20]

x_1^*=3.16246061572185; x_2^*=3.12833142812967;
x_3^*=3.09479212988791; x_4^*=3.06145059523469;
x_5^*=3.02792915885555; x_6^*=2.99382606701730;
x_7^*=2.95866871765285; x_8^*=2.92184227312450;
x_9^*=0.49482511456933; x_{10}^*=0.48835711005490;
x_{11}^*=0.48231642711865; x_{12}^*=0.47664475092742;
x_{13}^*=0.47129550835493; x_{14}^*=0.46623099264167;
x_{15}^*=0.46142004984199; x_{16}^*=0.45683664767217;
x_{17}^*=0.45245876903267; x_{18}^*=0.44826762241853;
x_{19}^*=0.44424700958760; x_{20}^*=0.44038285956317.

The algorithm is executed with the following settings: population size=200; amount of generations=150; *C*=100; the ratio of NS-NR procedures=15/1; crossover probability=0.95; Simulated Binary Crossover parameter n_c=1.0; mutation probability=0.05; polynomial mutation parameter n_m=30. The results are shown in figure 11-14 using a logarithmic scale for the horizontal axis, which shows the total violation *G*. From the figures it is observed how the initial population gradually approaches towards the optimal solution. It is emphasized that an iteration of the algorithm consists of 15 Pareto-ranking based generations, followed by one probabilistic selection based generation.

After 10 iterations the best feasible solution is found to be
$f(\boldsymbol{x})$= -0.793613533117088
The population is seen in figure 11. The independent variables of this solution take:

x_1=3.24832595081784; x_2=2.94319650443766;
x_3=2.94428354644506; x_4=3.02142730074793;
x_5=2.86945102101479; x_6=2.96442488220189;
x_7=0.526507749698735; x_8=0.429780319936723;
x_9=0.544135374090413; x_{10}=0.540324629305664;
x_{11}=3.12247385164555; x_{12}=3.04629476487622;
x_{13}=0.475892826530603; x_{14}=0.400468968498461;
x_{15}=0.525406871697624; x_{16}=0.363091228109451;
x_{17}=0.456317769218481; x_{18}=0.413066649730819;
x_{19}=0.466386058423425; x_{20}=0.536280452626657.

The peculiarity of the problem is essentially due to being highly non-linear, non-polynomial, and non-quadratic, -cubic, -quartic etc. the case being rather unconventional as to the examples subjected to evolutionary optimization and reported in the literature.

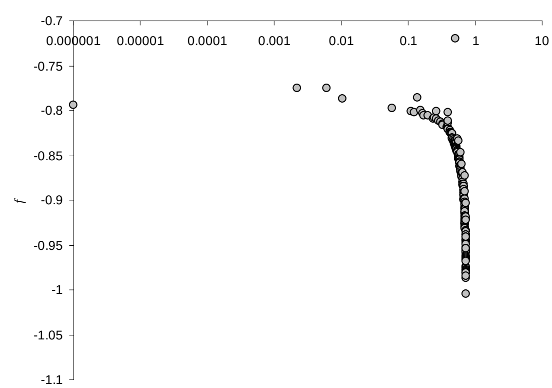


Fig. 11. Population after 10 iterations; horizontal axis shows the total violation *G* on a log scale.

After 20 iterations the best feasible solution is found to be
$f(\boldsymbol{x})$= -0.80305132174103
The population is seen in figure 12. The independent variables of this solution take:

x_1=3.15502583606141; x_2=3.11112176183396;
x_3=3.02496543675572; x_4=2.98747208109771;
x_5=2.9515112756444; x_6=2.89510918982729;
x_7=0.46796083643403; x_8=0.473668811347126;
x_9=0.467568074848906; x_{10}=0.452585498100958;
x_{11}=3.10462563793842; x_{12}=3.04573276503504;
x_{13}=0.471862973631331; x_{14}=0.463578991183557;
x_{15}=0.465680838811579; x_{16}=0.447391069763821;
x_{17}=0.469506617661979; x_{18}=0.42753345080416;
x_{19}=0.469472715928338; x_{20}=0.519950183966872.

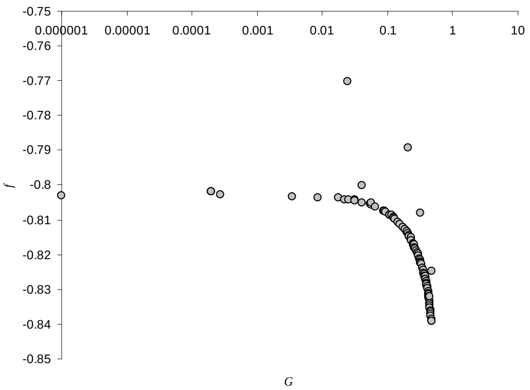


Fig. 12. Population after 20 iterations; horizontal axis shows the total violation G on a log scale.

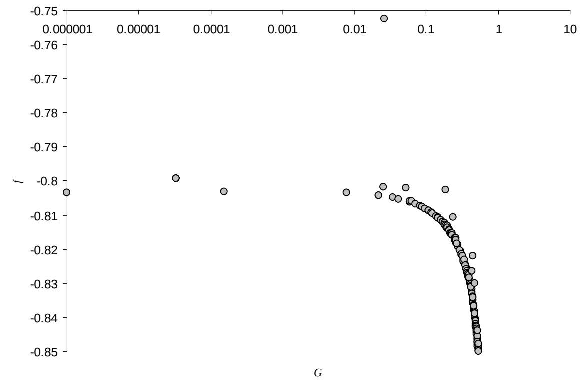


Fig. 14. Population after 60 iterations; horizontal axis shows the total violation G on a log scale.

After 30 iterations the best feasible solution is found to be
$f(\boldsymbol{x})= - 0.803340250367163$

The population is seen in figure 13. The independent variables of this solution take:

$x_1=3.1696117425466$; $x_2=3.09408201986905$;
$x_3=3.0172487986671$; $x_4=2.99495708426546$;
$x_5=2.95102962307473$; $x_6=2.89618000831499$;
$x_7=0.497409212169248$; $x_8=0.482812017517757$;
$x_9=0.465996025171434$; $x_{10}=0.452970326855209$;
$x_{11}=3.12293340944006$; $x_{12}=3.04402593262227$;
$x_{13}=0.484686923390343$; $x_{14}=0.462400174550483$;
$x_{15}=0.455413721826665$; $x_{16}=0.447678701325465$;
$x_{17}=0.457424900628494$; $x_{18}=0.436132029824826$;
$x_{19}=0.443064763267789$; $x_{20}=0.509337332371848$.

After 60 iterations the best feasible solution is found to be
$f(\boldsymbol{x})= -0.803340250367163$

The population is seen in figure 14. The independent variables of this solution take the same value as after 30 generations.

VI. CONCLUSIONS

A new approach for constrained optimization is presented, where the multiobjectivity of the problem is due to the Constraints. Conventionally, in a multi-objective constrained problem, with evolutionary search, the convergence is dominated by the constraints, if the number of constraints is

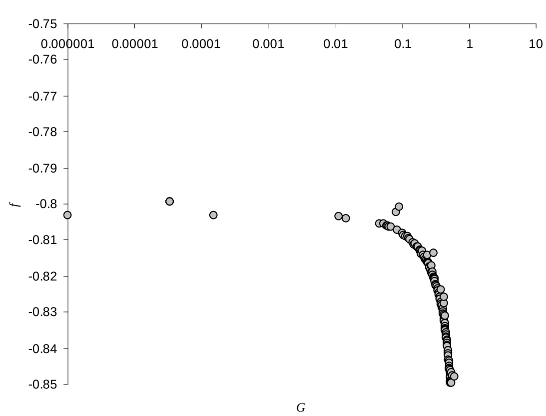


Fig. 13. Population after 30 iterations; horizontal axis shows the total violation G on a log scale.

high. This means, in the solution the optimization of the objective function is marginalized by the constraints. However, with the new method this undesirable situation is eliminated, and a clear improvement is achieved in a balanced manner. That is, during the search, both the objective and the constraints are equally stressed. The front is formed with advanced search operations, enabling a probabilistic nonlinear ranking, which is used for both NS and NR based tournament selection followed by elitism. For these operations an evolutionary probabilistic model of the random solutions is established. The model is used for an effective ranking procedure throughout the generations, yielding both robust and rapid convergence. The NR process of solutions is done always in a probabilistic scale, due to the adaptive feature of the probabilistic model, the outcomes of which are between zero and unity. This way the same precision is preserved, being independent of the level of convergence to the optimum. This means the method forms a dynamic "lens," the magnifying power of which is commensurate with the scale of convergence. This way convergence is accomplished accurately and systematically with precision, at any range allowed by machine or genotype coding precision. Relative to the conventional approach, the method shows outstandingly better performance as to precision, approaching to the solution without recourse to auxiliary supports like local search, memetic algorithm etc. The theory presented in this work is exemplified by a peculiar, highly-nonlinear, non-polynomial, non-quadratic etc. optimization problem for demonstration of the effectiveness of the methodology. This is a standard problem chosen from the literature for comparison of the results. We note that the results using the non-linear ranking developed in this work are very close to the best known optimum satisfactorily after few generations. Other examples are reported in the second part of this work, which is devoted to implementation and applications [10]. In both parts of the sequel, the reported results include not only the final outcomes but also the progress of the convergence throughout the optimization process, clearly showing the exact matching of the result with the theoretical considerations presented with a transparent convergence.

REFERENCES

[1] S. Das, S. S. Mullick, and P. N. Suganthan, "Recent advances in differential evolution - An updated survey," *Swarm and Evolutionary Computation,* vol. 27, pp. 1-30, 2016.

[2] K. V. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution - A Practical Approach to Global Optimization*: Springer, 2005.

[3] D. E. Goldberg, *Genetic Algorithms*. Reading, Massachusetts: Addison Wesley, 1989.

[4] C. A. C. Coello, D. A. Veldhuizen, and G. B. Lamont, *Evolutionary Algorithms for Solving Multiobjective Problems*. Boston: Kluwer Academic Publishers, 2003.

[5] K. Deb, *Multiobjective Optimization using Evolutionary Algorithms*: John Wiley & Sons, 2001.

[6] C. M. Fonseca, "An overview of evolutionary algorithms in multiobjective optimization," *Evolutionary Computation,* vol. 3, pp. 1-16, 1995.

[7] C. A. C. Coello, "An updated survey of Ga-based multi-objective optimization techniques," *ACM Computing Surveys,* vol. 32, pp. 109-143, 2000.

[8] K. Deb, "An efficient constraint handling method for genetic algorithms," *Computer Methods in Applied Mechanics and Engineering,* vol. 186, p. 28, 2000.

[9] A. C. A. Coello, "Use of a self adaptive penalty approach for engineering optimization problems," *Computers in Industry,* vol. 41, pp. 113–127, 2000.

[10] M. Bittermann, O. Ciftcioglu, and I. S. Sariyildiz, " Precision evolutionary optimization. Part II: Implementation and applications.," presented at the GECCO 2012, Philedelphia, 2012.

[11] S. Gass and T. Saaty, "The computational algorithm for the parametric objective function," *Naval Research Logistics Quarterly,* vol. 2, p. 7, 1955.

[12] L. Zadeh, "Non-scalar-valued performance criteria," *IEEE Trans. Automatic Control,* vol. 8, p. 2, 1963.

[13] K. Miettinen, *Nonlinear Multiobjective Optimization*. Boston: Kluwer Academic, 1999.

[14] Y. Y. Haimes, L. S. Lasdon, and D. A. Wismer, "On a bicriterion formulation of the problems of integrated system identification and system optimization," *IEEE Trans. Systems, Man, and Cybernetics,* vol. 1, p. 2, 1971.

[15] G. Bachman and L. Narici, *Functional Analysis*. New York: Dover, 2000.

[16] J. T. Oden and L. F. Demkowicz, *Applied Functional Analysis*: CRC Press, 1996.

[17] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multi-objective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation,* vol. 6, pp. 182-197, 2000.

[18] M. S. Bittermann and O. Ciftcioglu, "Precision Evolutionary Optimization Part II: Implementation and Applications " *The Journal of Cognitive Systems,* vol. 1, 2016.

[19] S. Koziel and Z. Michalewicz, "Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization," *Evolutionary Computation,* vol. 7, pp. 19-44, 1999.

[20] J. J. Liang, T. P. Runarsson, E. Mezura-Montes, M. Clerc, P. N. Suganthan, C. A. C. Coello, and K. Deb, "Problem definitions and evaluation criteria for the CEC 2006: Special session on constrained real-parameter optimization," *Journal of Applied Mechanics,* vol. 41, 2006.

Özer Ciftcioglu was born in Turkey. He received the PhD degree in Electrical Engineering from Istanbul Technical University (ITU) in 1976. He became associate professor in 1982 and full professor in 1988 at the same university.

Since 2009 he has been working at TU Delft in the Netherlands as Emeritus professor. He had been working at the same university from 1998 to 2009. From 1967 to 1998 he was affiliated with ITU as an academic member. His current research interests are Signal Processing, Computational Intelligence, and Computational Cognition.

Prof. Ciftcioglu is a senior member of IEEE since 2001.

S. Serhat Seker was born in Istanbul on July 2, 1959. He started his education at Mathematics Engineering Department of Istanbul Technical University (ITU) and graduated from Electrical and Electronic Engineering Faculty where he started his career as a research assistant at Electrical Power Engineering Department in 1985. He got his master degree at ITU's Nuclear Energy Institute and his doctorate at Electrical Engineering Division of the same university's Science and Technology Institute with his research titled as "Stochastic Signal Processing with Neural Network in Power Plant Monitoring."

Dr. Seker studied during his PhD thesis at Energy Research Centre of the Netherlands (ECN) with his scholarship provided from Netherlands Organization for International Cooperation in Higher Education (NUFFIC) and worked on signal analysis techniques there. He was titled as Assistant Professor and Associate Professor at ITU in 1995 and 1996 respectively. Also, he worked on industrial signal processing at Nuclear Engineering Department and Maintenance and Reliability Centre of the University of Tennessee, Knoxville-USA, by getting the scholarship from the Scientific and Technological Research Council of Turkey (NATO-TUBITAK B1) in 1997. He had many administrative duties at ITU in the previous years, including his vice dean position at Electrical and Electronic Engineering Faculty during 2001 - 2004. He was a department head in Electrical engineering department between 2004 and 2007. Also, he was dean of Technology Faculty and vice rector as well as the founding dean of Engineering Faculty at Kırklareli University between 2011 and 2013. Dr. Seker was dean of the Electrical and Electronics Engineering Faculty at ITU between 2013-2016.

Jelena Dikun was born in Klaipeda, Lithuania. She received P.B. from Klaipeda Business and Technology College in Electrical Engineering Faculty (2005-2009) and BSc from Klaipeda University in Electrical Engineering Faculty (2009 – 2011). She visited as a trainee Istanbul Technical University between October 21 and November 22, 2012 that provided by projects JUREIVIS and “Lithuanian Maritime Sectors' Technologies and Environmental Research Development”. Her areas of interest are the study of electric and magnetic fields.

Emine Ayaz received the BSc, MSc and PhD degrees from the Istanbul Technical University (ITU), Electrical Engineering Department, in 1993, 1997 and 2002 respectively. In 1999, she joined to the University of Tennessee, Nuclear Engineering Department and Maintenance & Reliability Centre to do research on accelerated aging studies of the electric motors. She is currently associate professor in Electrical Engineering Department of ITU. Her research interests are signal processing, soft computing, and condition monitoring in electric power systems.