THE JOURNAL OF COGNITIVE SYSTEMS

an international, peer-reviewed indexed, and open-access periodical

VOLUME 01
NUMBER 01
YEAR 2016
ISSN 2548-0650

www.dergipark.gov.tr/jcs

THE JOURNAL OF
COGNITIVE SYSTEMS

An international, peer-reviewed,
indexed, and open-access periodical

ISSN 2548-0650

GENERAL PUBLICATION DIRECTOR

S. Serhat Seker, Istanbul Technical University, Turkey

EDITOR-IN-CHIEF

Özer Ciftcioglu, Delft University of Technology, The Netherlands
Istanbul Technical University, Istanbul, Turkey

EDITORIAL BOARD

Michael S. Bittermann, Maltepe University, Istanbul, Turkey

Tahir Cetin Akinci, Istanbul Technical University, Turkey

SCIENTIFIC COMMITTEE

Abraham Lomi (Indonesia)

Belle R. Upadhyaya (USA)

Chandan Kumar Chanda (India)

Damian Jade Duff (Turkey)

Hakan Temeltas (Turkey)

Jelena Dikun (Lithuania)

Kunihiko Nabeshima (Japan)

Marija Eidukeviciute (Lithuania)

Murari Mohan Saha (Sweden)

Okyay Kaynak (Turkey)

Osman Nuri Ucan (Turkey)

Rituparna Datta (South Korea)

AIM & SCOPE

The journal publishes original papers in the field of Cognitive Systems, at large. It accepts contributions by research, as well as by review, and from the field of Cognitive Philosophy. Letter to the editor is also encouraged.

© 2016
THE JOURNAL OF
COGNITIVE SYSTEMS

Contact
Prof. Dr. Özer Ciftcioglu
Editor-in-Chief of The Journal of Cognitive Systems

Delft University of Technology, The Netherlands
Istanbul Technical University, Istanbul, Turkey

E-Mail
o.ciftcioglu@tudelft.nl
ozerciftcioglu@itu.edu.tr

URL
www.dergipark.gov.tr/jcs

THE JOURNAL OF
COGNITIVE SYSTEMS

VOLUME 01, NUMBER 01

DECEMBER 2016

CONTENTS

This page is intentionally left blank.

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:

ISSN 2548-0650 © 2016 The Journal of Cognitive Systems (JCS). Personal use is permitted, but republication/redistribution requires JCS permission. See www.dergipark.gov.tr/jcs for more information.

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 $0 \le w_i$ 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

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

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

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

where w_1=1, $w_{2i}=\mu_I$. 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 $f_l(\boldsymbol{x})$; subject to $f_j(x) \le \varepsilon_j$ for all $j=1,2,\ldots,k$, $j \ne l$; $x \in S$ where $l \in \{1,\ldots,k\}$. Naturally, inequalities can be converted to equalities by taking $\varepsilon_j=0$ for all $j=1,2,\ldots,k$, $j \ne l$.

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

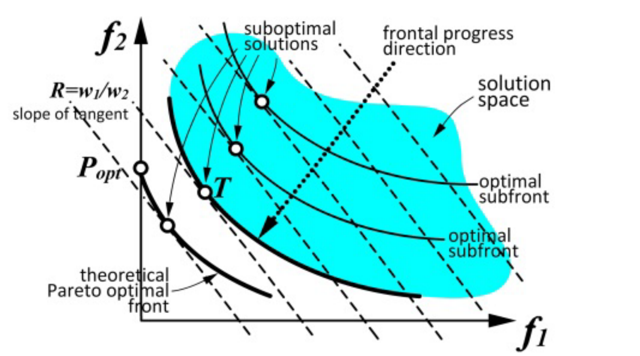


Fig. 1. Approach to the final optimal solution by means of constant penalty parameter *R*.

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.

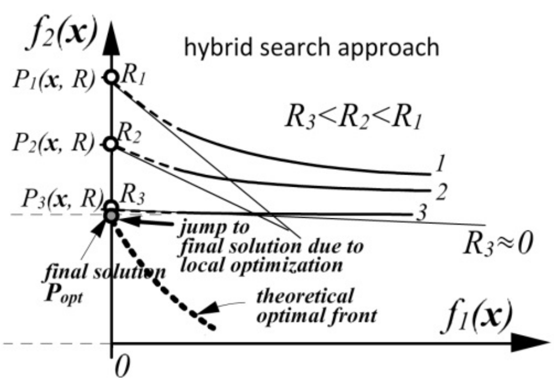


Fig. 2. Approach to the final optimal solution by means of penalty function approach, where R is the penalty parameter being estimated through curve fitting

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

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

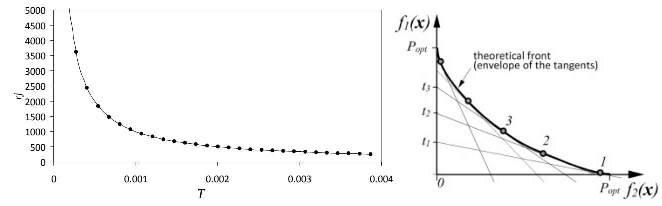
where t is the parameter. In (10), P_{opt} is the optimum solution where $f_2(\boldsymbol{x})= P_{opt}$ and $f_1(\boldsymbol{x})=0$. *From (10),* we write

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

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,

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

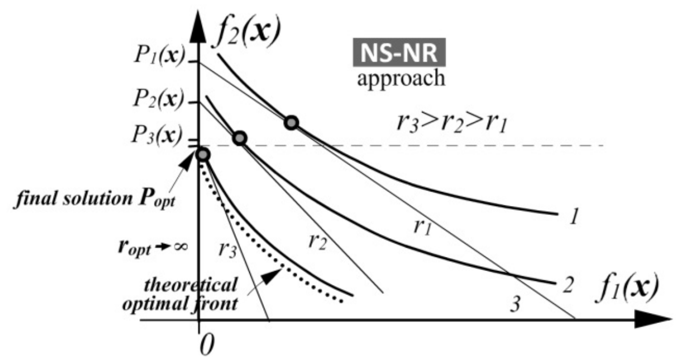


Fig. 4. NS-NR approach to the final optimal solution by means of penalty function approach; r is the penalty parameter.

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

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

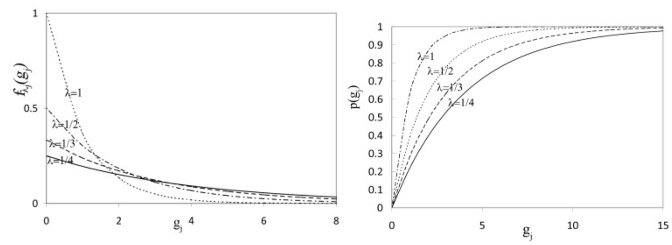
The mean value of the exponential pdf function is equal to λ_j^{-1}. During the evolutionary search $g_i(x)$ is a general form of violation which applies to any member s of the population although s is not explicitly denoted. However, in explicit form, we can write

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

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

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

that gives

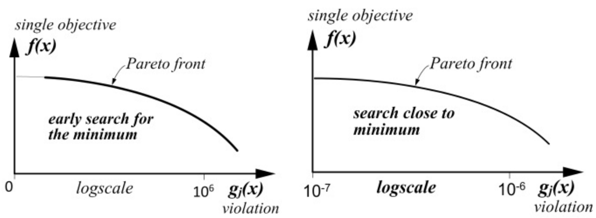
$$f_p(p) = 1 \tag{33}$$

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

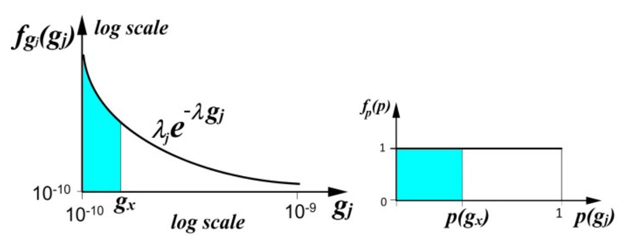
Adaptive zooming for ranking with precision is accomplished by accurate ranking the favourable solutions in the range zero and unity as probabilistic distances, even though the actual constraint values may be close to the optimal point as much as the computer precision can allow, say at the range of 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.



(a) (b)

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



(a) (b)

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

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

where $n_{bj}= ln2/\lambda_j$ 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; $n_{bj}=ln2/\lambda_j$ is the median of the exponential pdf as shown in figure 8b. For $n_{bj}=ln2/\lambda_j$, 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

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

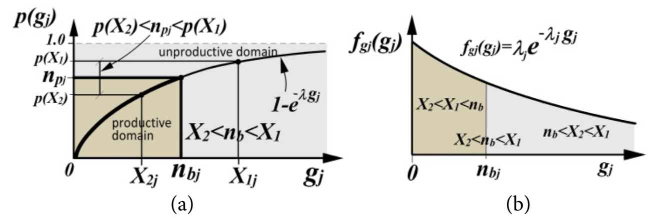


Fig. 8. Illustration of the threshold assessment for the tournament selection in both NS and NR procedures.

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 $n_{bj}=ln2/\lambda_j$. 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].

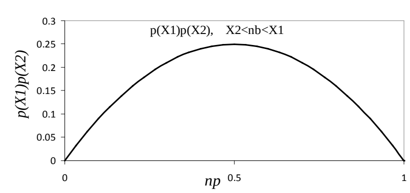


Fig. 9. Plot of the probability that two solutions occur on different sides of the thtreshold n_{bj} vs n_{pj}

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)

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

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

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

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.

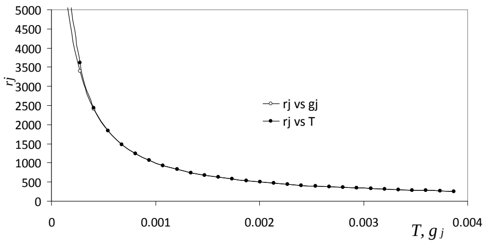


Fig. 10. Illustration of the *new* penalty parameter *r* as to probabilistic modeling: $r=(1-\exp(-\lambda g))/g$ and as to bi-objective formulation: $r=t/(P_{opt}-t)$

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).

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

subject to

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

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

The best known optimum is
$f(\mathbf{x}^*)$=-0.80361910412559 [20], and $f(\mathbf{x}^*)$=-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
$f(\mathbf{x})$= -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.

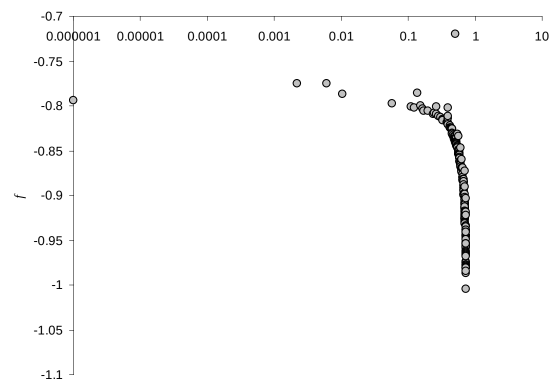


Fig. 11. Population after 10 iterations; horizontal axis shows the total violation *G* on a log scale.

After 20 iterations the best feasible solution is found to be
$f(\mathbf{x})$= -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.

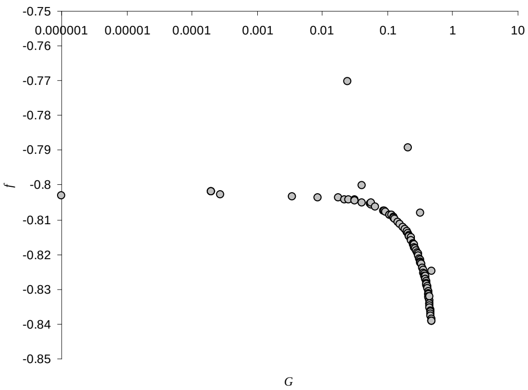


Fig. 12. Population after 20 iterations; horizontal axis shows the total violation *G* on a log scale.

After 30 iterations the best feasible solution is found to be
f(**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(**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

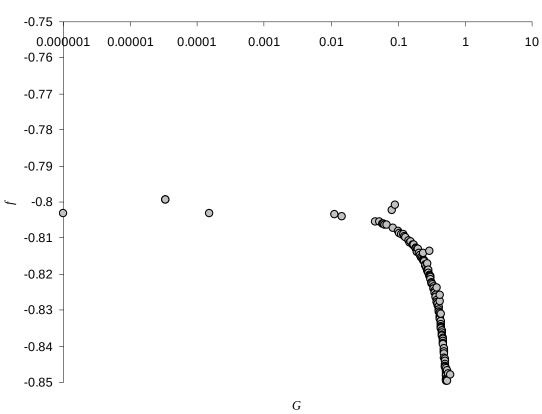


Fig. 13. Population after 30 iterations; horizontal axis shows the total violation *G* on a log scale.

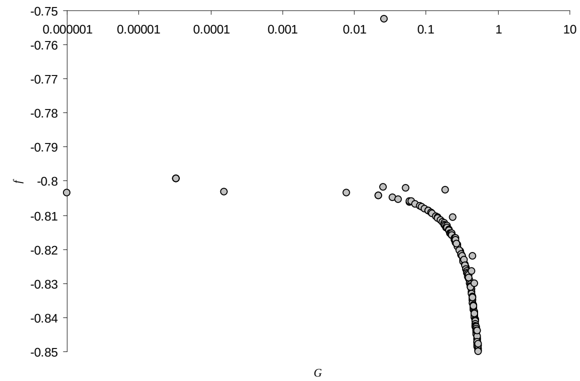


Fig. 14. Population after 60 iterations; horizontal axis shows the total violation *G* on a log scale.

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.

THE JOURNAL OF
COGNITIVE SYSTEMS
VOLUME 01 NUMBER 01 DECEMBER 2016

PRECISION EVOLUTIONARY OPTIMIZATION PART II: IMPLEMENTATION AND APPLICATIONS

Michael S. Bittermann[1], Tahir Cetin Akinci[2], Ramazan Caglar[2]

1 Maltepe University, Istanbul, Turkey
2 Istanbul Technical University, Istanbul, Turkey

Implementation and applications of a new approach to multiobjective optimization by evolutionary algorithms are presented. After non-dominated sorting for Pareto formation, a novel non-linear ranking is proposed during the fitness evaluation and tournament selection, as well as elitism. The non-linear ranking is based on a probabilistic model, which models the density of the genetic population throughout the generations by means of an exponential distribution. From this model, a robust probabilistic distance measure is established. The distance comprises a penalty parameter in an embedded form, which plays an important role for the convergence of the optimization process as it varies in an adaptive form during the generations in progress. Because of the embedded form, the penalty parameter is inherently tuned for every constraint, making the convergence, robust, fast, accurate, and stable. By the nonlinear ranking procedure, also the stiffness among the constraints is handled effectively. Convergence process is backed-up with an additional probabilistic threshold applied to the population, classifying them as productive and unproductive infeasible solutions. The details of the underlying theoretical work are presented in the first part of this sequel. The present work at hand describes the algorithmic implementation in detail, and the outstanding performance of the optimization process is exemplified by computer experiments. The problems used in the experiments are selected from the existing literature for the purpose of eventual benchmark comparisons.

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

I. INTRODUCTION

THERE IS CONTINUOUS growing interest in multi-objective evolutionary algorithms since their initial introduction some three decades ago. The algorithms are of interest in many diverse areas that may span diverse engineering science areas including the cognitive science. They are particularly suitable for the optimization tasks because they evolve simultaneously a population of potential solutions to the problem at hand, which allows one to search a set of favorable solutions in the form of an optimal front in a single run of the algorithm. Multi-objective optimization problems can be formulated in various ways depending on the problem at hand. One prominent example along that line is the constraint optimization [1], which is the subject matter of this work. In general multi-objectivity in optimization is a broad field in which much remains to be done in order to increase its effectivity in the diverse areas, where engineering applications take an important place [2]. The tutorials on evolutionary algorithm are widely available in the literature [3-5]. The updated research surveys on it are also available, e.g. [6, 7].

Since a multi-objective optimization can be formulated as a single objective problem with constraints, where the constraints are combined to be an additional objective subject to minimization, it is interesting to tackle the constraint optimization with single objective function as a general case. The method known as penalty function method is a commonly used method for constraint optimization. Following the penalty function method a solution is penalized, i.e. its fitness deteriorates when it violates constraints. This penalization is accomplished by adding a value to the objective function value in proportion to the amount of constraint violation, the proportionality factor being the penalty parameter. An evolutionary constrained optimization approach without penalty parameter was proposed by Deb in 2000 [8]. Due to the determination of the penality parameter during the search, Coello [9] proposed a self-adaptive penalty approach. Although introduction of penalty function for evolutionary multiobjective optimization problems is a general approach, the essential issue is the selection of the suitable penalty parameter which is dependent on each constraint of the penalty function. Therefore selection of a common penalty parameter becomes an oversimplification of the problem. As result of this, the approaches mentioned before leave a lot to be desired due to inadequate converge to the optimum while this is demanded. This is circumvented to some extent by using a classical optimization approach in combination with the evolutionary computation in order to converge the optimum matching the demands [1].

This paper addresses the multi-objective optimization as a bi-objective optimization where penalty function plays an important role. In this paper a new approach is proposed eliminating the need of classical constraint optimization next

Manuscript received September 13, 2016; accepted November 24, 2016.
Corresponding author: Michael S. Bittermann (E-mail: michaelsbittermann@maltepe.edu.tr).
Digital Object Identifier:

ISSN 2548-0650 © 2016 The Journal of Cognitive Systems (JCS). Personal use is permitted, but republication/redistribution requires JCS permission. See www.dergipark.gov.tr/jcs for more information.

to the evolutionary computation, yet providing outstanding convergence properties. In this approach a probabilistic model of the random solutions is used to derive a nonlinear distance measure that it is used for effective, i.e. robust ranking of genetic population members, and efficient, i.e. fast converging, solutions. The research is organized in two parts. The first part is presented in a theoretical framework with a demonstrative example afterwards [10]. In the second part, namely this part, 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, the formulation of general multi-objective optimization problem as constrained single objective problem is described. In section three probabilistic constraint handling is presented. In Section four, implementation of the probabilistic approach for non-linear ranking in an evolutionary algorithm is described. This is followed by a demonstrative computer experiment in section five, and conclusions.

II. METHOD FOR MULTIOBJECTIVE OPTIMIZATION

A. Weighting Method

The base of the problem formulation in this are the considerations known as *weighting method* [11-13]. In this method each objective is associated with a weighting coefficient and the weighting sum of the objectives is minimized. Thus, the multiple objective functions are converted into a single objective function. We assume that the weighting coefficients w_i are real numbers such that $0 \le w_i$ for all objectives i=1,….,k so that a weighting problem can be stated as

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

B. Weighting Method Formulated as Constrained Optimization

In the constraint handling in this work a single objective is involved which is subject to minimization. Therefore the problem can be stated as

$$\min \ f(\boldsymbol{x}) \ \text{subject to} \ g(\boldsymbol{x}) = [g_1(x), g_2(x), ..., g_m(x)]^T \le 0 \tag{2}$$

We assume that the feasible region is of the form

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

Considering that, the summation of the constraint violations is as another objective subject to minimization, the problem statement 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

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

Thus, the problem definition becomes explicitly,

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

With this formulation, the weighting method becomes appropriate to employ where w_1=1, w_{2i}=μ_i. We can formulate the multiobjective optimization as two objectives optimization which can be treated further a single objective with constraints, without deviating from generality. Such an approach is known as ε-*Constraint* method [13, 14].

C. Issues of the Penalty Function Approach

The problem statement given in (6) is written as

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

The function $g_j(\boldsymbol{x})$ is penalty function and the parameters R_j are the associated penalty parameters. Since each penalty parameter R_j indexed by the index parameter j is subject to identification, and this is a formidable task. To alleviate the issue, a common penalty parameter may be defined, so that (7) becomes

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

The selection of the penalty parameter R can be done in two ways:

1) Selecting a constant R. This case is illustrated in figure 1.

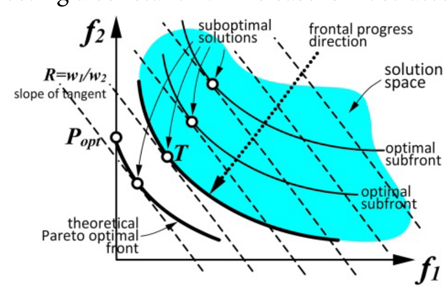


Fig. 1. Approach to the final optimal solution by means of penalty function approach; R is the penalty parameter

From the figure it is clear that, we can hope to converge to the tangent at the point T which is far from the optimum P_{opt}. Therefore a constant R is not a satisfactory strategy.

2) To determine a variable R, an extrapolation polynomial can be used, extrapolating the Pareto front. At the intersection of the polynomial and the $f_2(x)$ the slope of the tangent gives some estimate of R [1]. However, in this case R goes gradually zero tending to ignore the constraints. This is depicted in figure 2. Gradient based constrained local search has to be invoked to obtain the optimal point [1]. Evolutionary algorithm is used to

estimate a favorable starting point for the local search which makes the search very precarious.

It is to note that, with above consequences the need for the evolutionary algorithm becomes subject to discussion, as there is no point to expect that the population converges to the optimum. In essence the main machinery for optimization becomes the local search, where evolutionary optimization becomes merely a tool providing a favorable starting point for a non-evolutionary optimization.

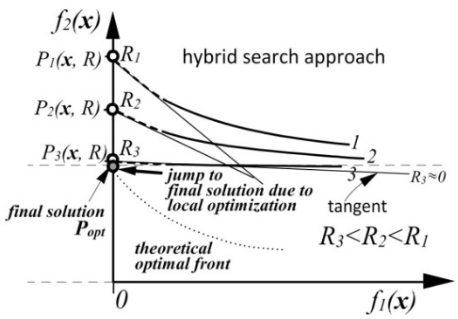


Fig. 2. Approach to the final optimal solution by means of penalty function approach; *R* is the penalty parameter

III. PROBABILISTIC APPROACH

A. PROBABILISTIC MODELING

As a new approach, we assume the problem formulation as a constraint optimization with single objective, so that in a general constrained optimization problem of the form

$$P(\boldsymbol{x}) = f(\boldsymbol{x}) + \sum_{i=1}^{J} \mu_j g_j(\boldsymbol{x}) \tag{9}$$

where $f(\boldsymbol{x})$ is the single objective function to be minimized; $g_j(\boldsymbol{x})$ is the violation of the j-th constraint, namely penalty function, μ_i is the associated parameter of the penalty function. At each generation during the evolutionary minimization process $g_j(\boldsymbol{x})$ is continually tried to be made to vanish. Considering the population density of solutions, this implies 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 (GA). 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. That is, the form of the density remains the same being independent of the range it models, and the exponential pdf is a unique density having this property. With this information peculiar to the subject matter of this research, we can confidently apply the exponential probability density function (pdf), which is given by

$$f_\lambda(y) = \lambda e^{-\lambda y} \tag{10}$$

where λ is the decay parameter. Denoting

$$y = g_j(x) \tag{11}$$

the pdf in (10) becomes

$$f_{g_j}(g_j) = \lambda_j e^{-\lambda_j g_j} \tag{12}$$

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

$$f_{g_j}(g_{j,s}) = \lambda_j e^{-\lambda_j g_{j,s}} \tag{13}$$

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

$$\lambda_j = 1/\bar{g}_j \tag{14}$$

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. 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 (12) is given by

$$p(g_j) = \frac{1}{g_j}\int_0^{g_j} e^{-\frac{g_j}{g_j}} dg_j = 1 - e^{-\frac{g_j}{g_j}} \tag{15}$$

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. Therefore in this work, in (9), μ_j is replaced by $Cr_j(g_j)$ in the form

$$C\,r_j(g_j) = \mu_j(g_j) \tag{16}$$

So that (9) becomes

$$P(\boldsymbol{x}) = f(\boldsymbol{x}) + C\sum_{i=1}^{J} r_j(g_j) g_j(\boldsymbol{x}) \tag{17}$$

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 [10]; r_j is a *new* penalty parameter which is a function of g_j. In (17), $r_j(g_j)g_j$ is replaced by $p(g_j)$, in the form

$$r_j(g_j)g_j = p_j(g_j) \qquad (18)$$

so that (17) becomes

$$P(\boldsymbol{x}) = f(\boldsymbol{x}) + C\sum_{i=1}^{J} p_j(g_j(\boldsymbol{x})) \qquad (19)$$

In view of (18), r_j is given by

$$r_j = f(g_j) = p_j(g_j)/g_j \qquad (20)$$

The plot of r_j vs g_j is shown in figure 3, and its variation during the evolutionary search as to the Pareto optimal front is shown in figure 4.

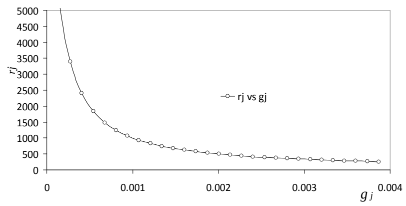


Fig. 3. Illustration of the *new* penalty parameter r as to probabilistic modeling: $r=(1-\exp(-\lambda g))/g$

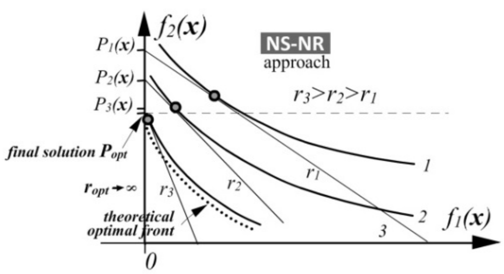


Fig. 4. Approach to the final optimal solution by means of penalty function approach; r is the penalty parameter.

IV. IMPLEMENTATION OF THE EVOLUTIONARY ALGORITHM

In the probabilistic formulation of a constraint optimization problem, the function subject to minimization is given by

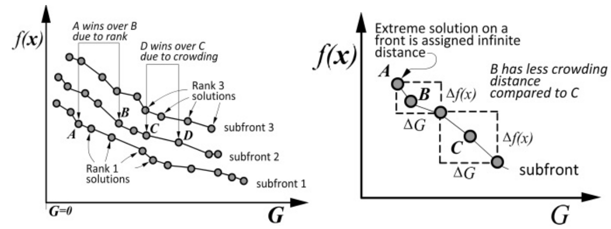
$$P(g_j, \boldsymbol{x}) = f(\boldsymbol{x}) + C\sum_{j=1}^{J} p\left(g_j(\boldsymbol{x})\right) \qquad (21)$$

where J is the number of constraints; C is a common constant. The probability $p(g_j)$ controls the penalty parameter, $\mu_i(g_j)$ which is absorbed in $p(g_j)$ in the form of r_j. The penalty parameter $\mu_i(g_j)$ varies theoretically between zero and infinity, while $p(g_j)$ varies between zero and unity.

A. Stage one: Non-dominated Sorting (NS)

As a first step in the algorithm, the multi-objective optimization problem is converted into a two-objective problem. The second objective subject to minimization is the summation of the violations. During the NS part of the algorithm we are considering G as second objective, i.e. the sum of the violations g_j and not the sum of the probabilities $p(g_j)$. The reason for that is that, as a first step the algorithm should establish a Pareto front in the bi-objective space, and the bounded range of p-space as unity, i.e. $0 \le p(g_j) \le 1$ implies a tendency for aggregation in the space formed by $f(\boldsymbol{x})$ and $p(g_j)$.

For the Pareto-front formation in the first step, the selection among the solutions is based on binary tournament selection using non-dominated sorting (NS) and crowding [15]. It is noted that this procedure is applied for infeasible solutions exclusively, i.e. solutions where $G<0$. Solutions are sorted with respect to the Pareto subfront they belong to, and assigned a Pareto rank index accordingly. This is seen from figure 5a. The crowding computation is illustrated in figure 5b for two solutions B and C, where solution C is preferred in a tournament due to larger crowding distance for C. The length of the cuboid around a solution is compared among the solutions on the same subfront,



(a) (b)

Fig. 5. Non-dominated sorting based selection among the infeasible solutions (a); Crowding distance computation (b)

and a solution with greater distance will be preferred over a solution with smaller distance. This is in order to avoid aggregation of solutions in the objective space, i.e. to reach a front with uniform density of solutions. Solutions at the extremity of a Pareto rank will be assigned infinite crowding distance, so that they will always prevail over other solutions on the same rank. This is to ensure that the sizes of the sub fronts remain large during the ranking-based front formation. Solutions in a tournament will be evaluated depending on the condition given by

$$\sum_{j=1}^{J} p(g_j) < n_{pj}\, J \qquad (22)$$

where J is equal to the number of constraints, and n_{pj} denotes a probability threshold, above which a solution is deemed *unproductive* among the infeasible solutions, and below which

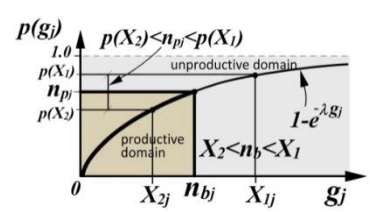


Fig. 6. Sketch for the selection procedure during non-dominanted sorting (NS) based tournament

a solution is deemed *productive*. It has a counterpart in the violation domain denoted by n_{bj}. This is seen in figure 6.For the condition in (22) three possible outcomes can occur:

1. In case both solutions fulfill condition (22), i.e. both solutions are in the productive domain, the solutions are compared with respect to their rank. The solution with

lower rank wins the tournament. In case they are on the same rank, the solution with greater crowding distance wins the tournament. The crowding distance is computed as seen in figure 5b [15].

2. In case both solution do not fulfill condition (22), i.e. both belong to the unproductive domain, then the solution, whose sum of $p(g_j)$ is smallest, wins the tournament without considering rank or crowding distance. This is to favor the solution among the two unproductive one, which is nearer to the productive domain.
3. In case one solution fulfills (22), while the other one does not, then she solution in the productive domain wins the tournament over the other one, without considering rank or crowding information. This case is shown in figure 6, where the violation in the productive domain is denoted by X_{2j} and its counterpart is X_{1j}.

Optimal selection of the threshold, n_{pj} or n_{bj} is explained in another publication, where the optimum value is identified to be 0.5 [ref. paper 1 JCS]. The functionality of (22) is especially due to case 3, as it increases the pressure, i.e. increased number of productive chromosomes, towards the feasible region. It is noted that the location of the boundary parameter n_p implies a fixed location in the $p(g_j)$-dimension, whereas in g_j-dimension the location of the boundary generally changes from generation to generation due to changing mean values.

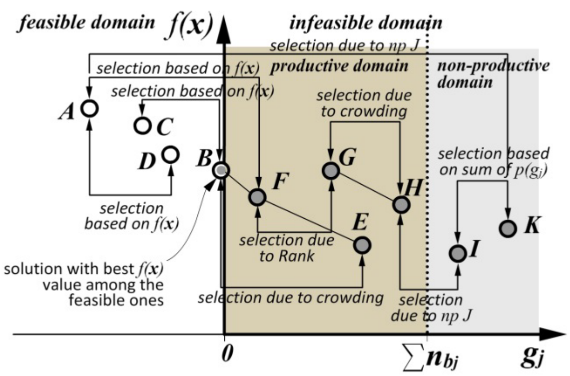


Fig. 7. Sketch for the tournament selection during NS

The possible comparison criteria and outcomes from binary tournaments mentioned above are exemplified in figure 7. During the search process feasible solutions may arise. In the binary tournament selection, when a feasible solution is selected together with an infeasible one, e.g. *A* and *F*, or two feasible ones are selected, e.g. *A* and *D* the comparison between the solutions is based on the values of $f(\boldsymbol{x})$ exclusively, i.e. without considering the violation information or rank. This means the winner of the tournament is the solution among the two that has lower value of $f(\boldsymbol{x})$, i.e. *F* wins over solution *A*, and in the same way *D* wins over *A*. Excluding the violation information is done, since for the feasible region the summation of the constraint violations is not defined. Namely the original optimization problem is to find a solution that minimizes $f(\boldsymbol{x})$, while the constraints are not violated, i.e. there is no need for reaching solutions within the feasible region away from the feasibility boundary. When two solutions from the productive domain are in a tournament, e.g. *F* and *G*, then *F* wins over *G* due to the lower rank of *F*. When a solution from the productive domain is in a binary tournament with a solution from the non-productive domain, e.g. solutions *H* and *I* in the figure, then *H* wins over *I*. And finally, when among two solutions from the non-productive domain, e.g. *I* and *K*, then *I* wins over *K*, as the former is nearer to the boundary separating the productive and non-productive domains. It is noted that by means of the distinction between productive and non-productive solutions, the probabilistic considerations are introduced to the conventional non-dominated sorting algorithm.

After the tournament selection the genetic operators are applied and a new population is created. In the present implementation simulated binary crossover [16] and polynomial mutation [17] are used for this procedure. When the new generation is formed an elitism concept is applied [15] in a modified form in this work, seen from figure 8. The new generation is combined with the previous one, and thereafter the infeasible solutions are sorted based on their rank and the feasible solutions based on their $f(\boldsymbol{x})$ values. The feasible

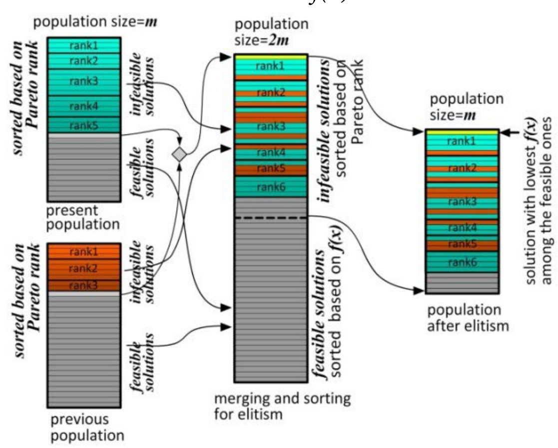


Fig. 8. NS based elitism

solutions with the lowest $f(\boldsymbol{x})$ values are used to fill up remaining places in the elitist population. The feasible solution with the lowermost value of $f(\boldsymbol{x})$ is put to the uppermost place in the population after elitism. This solution is marked in yellow in figure 8.

B. STAGE TWO: NON-LINEAR RANKING (NR)

The NS algorithm described above is repeated for a number of generations, for example four generations, so that the Pareto front sufficiently develops. Thereafter a non-linear ranking procedure based on the probabilistic considerations described above is employed as follows. During the tournament selection process, for two infeasible solution from the productive domain the value $P(g_j, \boldsymbol{x})$ in (2) is used to determine the winner of the tournament. In this procedure, clearly, a solution with lower P value is preferred over the solution with a larger P value. If a solution in the tournament belongs to the non-productive domain, then the same consequences apply as in the NS tournament. Namely, productive solutions win over non-productive solutions, and among non-productive solutions, the solution which is nearest to the productive domain wins.

Possible outcomes during the non-linear ranking procedure are exemplified in figure 9.

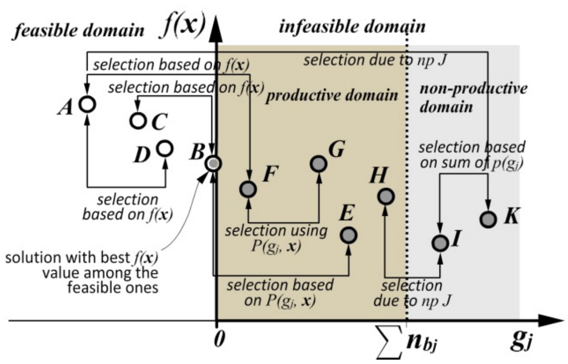


Fig. 9. Sketch of the tournament selection during NR

For instance, in the figure solution B represents the best solution among the feasible ones. When this solution is in a tournament with in infeasible solution from the productive domain, e.g. solution E, the winner of the tournament is obtained using $P(g_j, \boldsymbol{x})$. That is solution B is considered as if it were an infeasible one for this comparison, so the chance B remains in the population is increased. For solutions from the productive domain, as $P(g_j, \boldsymbol{x})$ is a summation of function value $f(\boldsymbol{x})$ and summed up values of $p(g_j)$, population members that have a low function value and at the same time small sum of $p(g_j)$ are favored in the selection process. A solution having a low summation of $p(g_j)$ means that this solution has the unusual property that it violates several constraints with an extraordinarily low amount, when considered in perspective with the average violations of the respective constraints. In contrast to the Pareto-ranking based algorithm exercised before, the probabilistic selection mechanism will not permit solutions with low function value to remain in the population, provided the coefficient C is selected large enough. The important implication of the NR tournament selection is assigning a commensurate right penalty parameter for every constraint, and even for each population member, where thepenalty parameter is embedded in the non-linear distance function [10]. By means of this, the robustness and precision of the algorithm is guaranteed, together with the high stability of the search process. After the non-linear ranking based tournament selection, $P(g_j, \boldsymbol{x})$ is used during an elitism procedure, as seen in figure 10. From the figure it is noted that in the sorting step for the elitism the infeasible solutions are sorted based on their $P(g_j, \boldsymbol{x})$ values. Generally the mean values for the different constraints of two consecutive generations being merged for elitism differ, and it is generally expected that the mean values improve from generation to generation. In order to ensure accurate convergence, in this implementation for the sorting procedure during the NR elitism $P(g_j,\boldsymbol{x})$ is obtained using the mean value of the respective generation when the chromosome was created. This way the convergence is slowed down in order to ensure that the solutions from the past generation will also have significant influence in the ensuing generation. This is in order to maintain diversity during the search and carefully target the minimum being approached with the population.

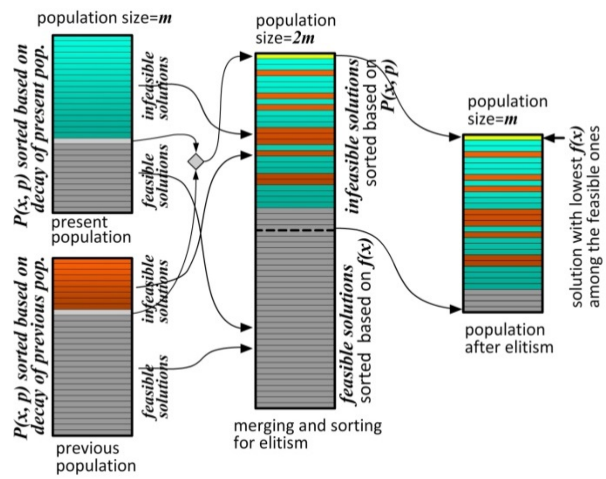


Fig. 10. NR based Elitism

V. COMPUTER EXPERIMENT

Computer experiments have been carried out using two optimization problems from the literature.

A. PROBLEM I

The following problem is due to Hock and Schittowski [18]. It is given by (23)-(25).

$$f(\boldsymbol{x}) = (x_1-10)^2 + 5(x_2-12)^2 + x_3^4 + 3(x_4-11)^2 + 10x_5^6 + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7 \quad (23)$$

where the ranges for the independent variables are given by

$$-10 < x_i < 10, i = 1,\ldots,7 \quad (24)$$

Subject to:

$$\begin{aligned} g_1(\boldsymbol{x}) &= -127 + 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 \le 0 \\ g_2(\boldsymbol{x}) &= -282 + 7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 \le 0 \\ g_3(\boldsymbol{x}) &= -196 + 23x_1 + x_2^2 + 6x_6^2 - 8x_7 \le 0 \\ g_4(\boldsymbol{x}) &= 4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7 \le 0 \end{aligned} \quad (25)$$

It consists of a single objective with four constraints, subject to minimization. The best known optimum is located at

$f(\boldsymbol{x}^*)= 680.630057374402$

The corresponding best known variable values are x_1^*=2.330499351474051 74; x_2^*=1.95137236847114592; x_3^*=-0.477541399510615805; x_4^*=4.36572624923625874; x_5^*=-0.624486959100388983; x_6=5; x_7^*=1.03813099410962173; x_8^*=1.5942266780671519.

The algorithm is executed with the following settings: population size=200; amount of generations=70; *C*=100000; the ratio of NS-NR procedures=4/1; crossover probability=0.9; mutation probability=0.05. The results are shown in figure 11-13 using a logarithmic scale for the horizontal axis, which shows the sum of the violations g_j denoted by *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 four Pareto-ranking based generations, followed by one probabilistic selection based generation. From figures 11-13 it is observed that the search process continues to yield solutions near to the optimal point. From the results it is noted how the initially scattered population gradually approaches as a connected front towards the optimal solution. The search maintains the pressure towards the feasible region throughout the search process and arrives at the feasible region with a large amount of potential solutions near to the optimum. This manifests the robustness of the approach.

After 10 iterations the best feasible solution is found to be $f(\mathbf{x})$= 681.776930738684.

This solution is near to the optimum, namely at a distance 1.68 promille from the best known optimum.

The population is seen in figure 11.

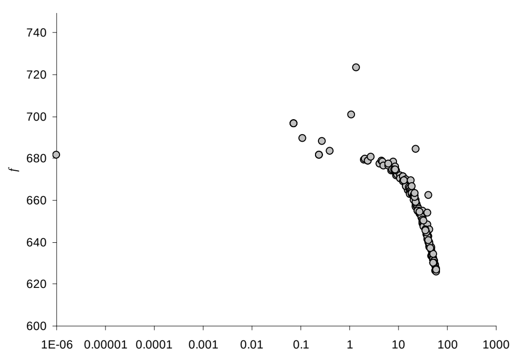


Fig. 11. Population after the 10-th iteration

The independent variables of this solution take:

x_1=2.32189959894901; x_2=1.95533366880135; x_3=0.0913466483171242; x_4=4.31676277251481; x_5=-0.462500971507716; x_6=1.04611582287531; x_7=1.59865097668138.

After 30 iterations the best feasible solution is found to be

$f(\mathbf{x})$= 680.67949252499.

The population is seen in figure 12. The independent variables of this solution take:

x_1=2.32743347740407; x_2=1.9576387118545; x_3=-0.503457841417583; x_4=4.34872456501762; x_5=-0.612760668700169; x_6=1.0244876812099; x_7=1.58909845884555.

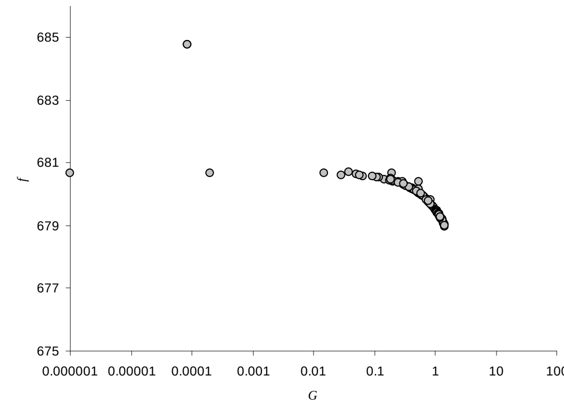


Fig. 12. Population after the 30-th iteration

After 70 iterations the best feasible solution is found to be

$f(\mathbf{x}^*)$= 680.632527938176.

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

x_1=2.33064474976019; x_2=1.95388009157449; x_3=-0.469607706232811; x_4=4.35926347613402; x_5=-0.62611714120937; x_6=1.03074889097774; x_7=1.58906253465783.

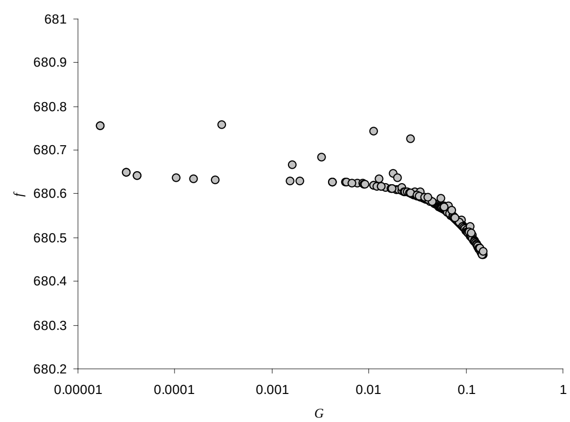


Fig. 13. Population after the 70-th iteration

B. PROBLEM II

The following problem is due to Floundas and Pardalos [19]. It consists of a single objective with two constraints, subject to minimization. The best known optimum is located at

$f(\mathbf{x}^*)$= *-6961.81387558015*

The corresponding best known variable values are x_1^*=14.09500000000000064; x_2^*=-0.138032130213039.

The problem is given by (26)-(28).

$$f(\boldsymbol{x}) = (x_1 - 10)^3 + (x_2 - 20)^3 \quad (26)$$

where the ranges for the independent variables are given by

$$13 < x_1 < 100;\ 0 < x_2 < 100 \quad (27)$$

subject to:

$$\begin{aligned} g_1(\boldsymbol{x}) &= -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 \le 0 \\ g_2(\boldsymbol{x}) &= (x_1 - 6)^2 + (x_2 - 5)^2 - 82.81 \le 0 \end{aligned} \quad (28)$$

The algorithm is executed with the following settings: population size = 200; amount of generations=100; C=100000; the ratio of NS-NR procedures=10/1; crossover probability=0.9; crossover parameter n_c=1.0; mutation probability=0.05; mutation parameter n_m=30. The results are shown in figures 14-16 using a logarithmic scale for the horizontal axis, which shows G being the total sum of the violations g_j.

After 30 iterations the best feasible solution is found to be

$f(\boldsymbol{x})$= -6944.7266618604.

The population is seen in figure 14. The independent variables of this solution take: x_1=14.1026225766318; x_2=0.858143925111059.

After 50 iterations the best feasible solution is found to be

$f(\boldsymbol{x})$= -6952.4044222655.

The population is seen in figure 15. The independent variables of this solution take: x_1= 14.0992588088961; x_2= 0.851316093914925.

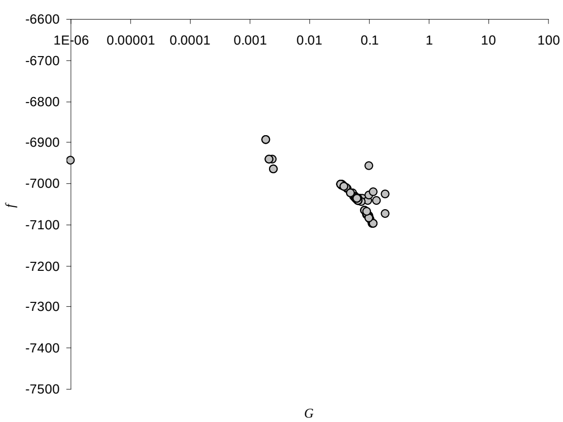


Fig. 14. Population after the 30-th iteration

After 100 iterations the best feasible solution is found to be

$f(\boldsymbol{x})$= -6961.75770743364.

The population is seen in figure 16. The independent variables of this solution take: x_1=14.095023241862; x_2=0. 0.843010744010595.

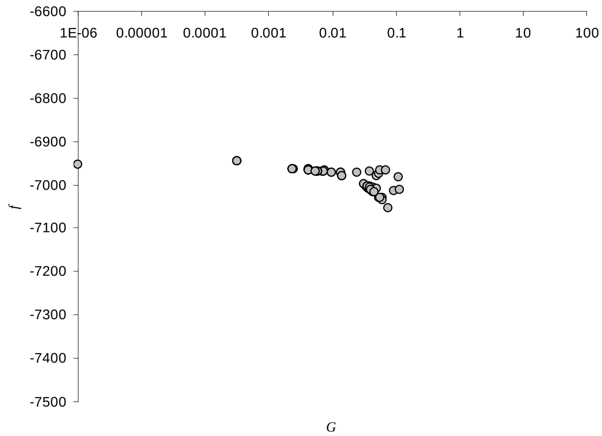


Fig. 15. Population after the 50-th iteration

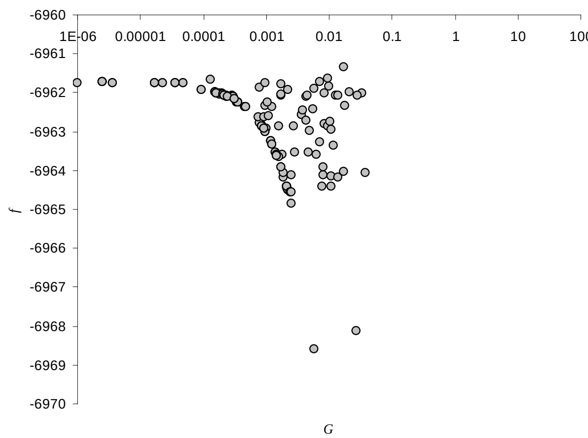


Fig. 16. Population after the 100-th iteration

VI. CONCLUSIONS

A new approach for multiobjective evolutionary optimization problem is presented. Conventionally the problem is handled in the form of single objective and the sum of constraints. However noting that in the optimal front formation the essential optimization progress is focused on the constraints where sum of a number of objectives are involved, the single objective is minimally attended yielding poor progress attached to it. As result conventionally in this problem formulation evolutionary computation has to be supported by auxiliary local search algorithms. By means of the new methodology a marked improvement is achieved for bi-objective formulation, i.e. for a single objective with constraints. Next optimal front formation during the search, also evolutionary minimization of the single objective is carried out in alternating sequence. By doing so, a balanced optimal search is established between the objectives forming the constraints and the single objective. The result is a markedly effective front for advanced search operations paving the way for a probabilistic nonlinear ranking used for both nonlinear tournament selection and nonlinear elitism. For these operations evolutionary probabilistic model for the random solutions is established for both robust and rapid convergence by means of effective ranking procedure throughout the generations, so that the results are not precarious. Based on this dynamic model, ranking the solutions is done always in a probabilistic scale, namely between zero and one preserving the same accuracy being independent of the level of convergence to the optimum; namely the method forms a dynamic "lens" whose magnifying power is commensurate with the scale of convergence. This allows accurate monitoring of convergence ensuring rapid convergence with precision. By the nonlinear ranking procedure, also the stiffness among the constraints is handled effectively by a commensurate model parameter, each of which is tuned for each individual constraint. The method showed outstanding performance as to robustness, precision, accuracy, and stability. Referring to the reported researches in the literature, a marked feature of the present algorithm is, that it approaches to the optimum in the same range of reported accuracy of the results without recourse to any auxiliary support like local search, memetic algorithm etc. that they make the search process dominated by the classical optimization methods rather than evolutionary. The performance of the algorithm is exemplified by means of two standard problems chosen from the literature for the comparison of the results. Another example is reported in another paper devoted to the theory underlying this work [ref. paper 01]. The reported results include not only the final outcomes, but also the progress of the convergence throughout the optimization process. This not only marks the effectiveness of the method proposed here, but also exhibits a transparency of the evolution throughout the generations.

REFERENCES

[1] K. Deb and R. Datta, "A fast and accurate solution of constrained optimization problems using a hybrid bi-objective and penalty function approach," presented at the IEEE Congress on Evolutionary Computation (CEC), Barcelona 2010.

[2] K. Deb, *Optimization for Engineering Design*. new Delhi: PHI, India, 1995.

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

[10] O. Ciftcioglu, M. S. Bittermann, and I. S. Sariyildiz, "Precision Evolutionary Optimization - Part I: Nonlinear Ranking Approach," presented at the GECCO 2012, Philadelphia, 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] 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.

[16] K. Deb and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex Systems,* vol. 9, pp. 115-148, 1995.

[17] K. Deb and M. Goyal, "A combined genetic adaptive search (GeneAS) for engineering design," *Computer Science and Informatics,* vol. 26, pp. 30-45, 1996.

[18] W. Hock and K. Schittkowski, "Test Examples for Nonlinear Programming Codes. ," in *Lecture Notes in Economics and Mathematical Systems*, ed Berlin: Springer-Verlag, 1981.

[19] C. Floundas and P. Pardalos., *A Collection of Test Problems for Constrained Global Optimization,* vol. 455. Berlin, Germany: Springer-Verlag, 1987.

Michael S. Bittermann was born in Germany in 1976. He graduated with the highest honors as MSc in Architecture from Delft University of Technology (TU Delft), The Netherlands, in 2003. He received his PhD degree in Design Informatics with the highest honors from the same university in 2009. Bittermann was distinguished as one of the five eminent researchers among all 2008-2009 PhD graduates of TU Delft in the form of the Young Researcher Fellowship of Delft University of Technology (80k€). He joined Maltepe University, Istanbul, Turkey in 2013, where he contributed to the research project titled 'Design by Cognition and Comprehension,' funded by the Scientific and Technological Research Council of Turkey. His research interest is computational cognition.

Tahir Cetin Akinci received the B.Sc. degree in Electrical Engineering. His master's and Ph.D. degrees, Institute of Pure and Applied Science from Marmara University, Istanbul, Turkey. He is an associate professor in the Department of Electrical Engineering at Istanbul Technical University (ITU), Istanbul, Turkey. His research interests are signal processing, data mining, intelligent systems, ferroresonance phenomenon, artificial neural networks, and renewable energy sources.

Ramazan Caglar received the B.Sc., and M.Sc. diploma in Electrical Engineering from the Istanbul Technical University in 1984 and 1987 respectively, and Ph.D. degrees in Graduate School of Science, Engineering and Technology of the same university in 1999. He accepted as Postdoctoral Researcher by the School of Electric and Computer Engineering at Georgia Institute of Technology, Atlanta, USA in 2001. He has conducted his research activities for two years in this university. His research interests are power system modeling, reliability of electric power system, the effects of deregulation on the electric power system, power transmission pricing, distributed generations, and fault diagnosis, risk management and signal processing.

In 1984 he joined the Faculty of Electrical Engineering, Istanbul Technical University as a research assistant, where he is presently an Assoc. Professor. He is active in teaching and research in the general modeling, analysis, and control of power systems. His has published article in some referred journals, and conference proceedings.

Dr. Caglar is a member of IEEE, IEEE-PES, a member of CIGRE, a member of Turkish National Committee on Illumination (ATMK), and a National Chamber of Turkish Electrical Engineering (EMO).

THE JOURNAL OF
COGNITIVE SYSTEMS
VOLUME 01 NUMBER 01 DECEMBER 2016

PROBABILISTIC SORTING FOR EFFECTIVE ELITISM IN MULTI-OBJECTIVE EVOLUTIONARY ALGORITHMS

S. Serhat Seker[1], Michael S. Bittermann[2], Ramazan Caglar[1], Rituparna Datta[3]

[1] Istanbul Technical University, Istanbul, Turkey
[2] Maltepe University, Istanbul, Turkey
[3] Graduate School of Knowledge Service Engineering, Korea Advanced Institute of Science and Technology (KAIST), 291 Daehak-ro, Yuseong-gu, 305-701, South Korea

One of the essential points in the evolutionary algorithms is the rank determination for the genetic population members. In this respect a new approach is presented, which is a probabilistic sorting for effective elitism and ensuing improved and robust convergence. This is achieved by an adaptive probabilistic model representing the commensurate probability density of the random solutions throughout the generations that it yields a probabilistic distance measure which is nonlinear with respect to the range of solutions as to their location in the objectives space. The implementation of the theoretical results leads an effective evolutionary optimization algorithm accomplished in two stages. In the first stage linear non-dominated sorting, tournament selection and elitism is carried out in objective space. In the second stage the same is executed in a transformed objective space, where probabilistic distance measure for ranking prevails. The effectiveness of the method is exemplified by a demonstrative computer experiment. The problem treated is selected from the existing literature for comparison, while the experiment carried out and reported here demonstrates the marked performance of the approach. The experiment complies with the theoretical foundations, so that the robust and fast convergence with precision as well as with accuracy is accomplished throughout the search up to 10-10 range or beyond, limited exclusively by machine precision.

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

I. INTRODUCTION

COMPUTATIONAL cognition makes use of the evolutionary optimization algorithms due to the decision-making process in the cognition. This is especially important in the action and communication stage of cognition. This work describes a research, which provides an effective method to enhance the effectiveness of evolutionary optimization algorithms, and consequently improve the cognition process. Evolutionary algorithms are powerful heuristic computations for multiobjective optimization problems. Their various forms of utilization are ubiquitous and they are reported regularly in the literature, e.g. [1, 2]. Some text book are available e.g. [3-5] that one can approach to master the topic. During the last decades evolutionary algorithms received growing interest, since they proved to be important tools for optimization. Added to that, they also proved to be effective in constraint optimization problem solving as the modern technological application areas imposes limitations on the solutions. The conventional constrained optimization methods generally use methods based on various penalty functions. Penalty function methods are generic but care has to be exercised to use the penalty parameter in a measured way to keep the balance between the constraints and the objective to avoid false optima and infeasible solutions. A strategy that does not use penalty parameter in evolutionary constrained optimization was proposed by Deb in 2000 [6, 7]. Although the penalty parameter can be kept constant during the search process, a better approach is to use a variable penalty parameter, which is adapted to the progress of the convergence, providing an effective approach to the optimum in the decision variable space. In this respect Coello proposed a self-adaptive penalty approach [8]. By doing so, also the evolutionary concept is clearly demonstrated. Conventionally in the penalty function approach, the constrained optimization problem is a search of the best compromises of the objective value and constraint satisfaction. Due to this construction, net result is the unsatisfactory convergence properties which are deemed to be repaired by some additional methods borrowed from classical optimization methods which are collectively addressed as 'local search' methods. One of the essential components in evolutionary algorithms is the rank determination for the individuals. In this work, this issue is addressed by means of probabilistic distance measure which is used for probabilistic sorting and effective elitism by a nonlinear ranking. The method provides a kind of 'mathematical lens,' so that at any stage of convergence the level of rank resolution remains the same that it leads systematic, smooth convergence to the optimum without recourse to additional methods which are collectively regarded as 'local search' methods.

The present work addresses the conversion of a single objective constrained optimization problem into a multiobjective, unconstrained optimization together with a penalty function. In this form, it is a bi-objective optimization problem. Each of the constraints has its own penalty

Manuscript received September 9, 2016; accepted December 7, 2016.
Corresponding author: Michael S. Bittermann (E-mail: michaelsbittermann@maltepe.edu.tr).
Digital Object Identifier:

ISSN 2548-0650 © 2016 The Journal of Cognitive Systems (JCS). Personal use is permitted, but republication/redistribution requires JCS permission. See www.dergipark.gov.tr/jcs for more information.

parameter. For each constraint a probabilistic model of the random solutions is used to derive a nonlinear distance measure. This measure is used for the genetic algorithm, to rank the population members for efficient, i.e. fast convergence. In this form it is a constraint optimization problem. This is the new approach proposed in this paper, for robust and stable solutions. The method is implemented by a computer program developed for this research working based on non-dominated sorting (NS) and non-linear ranking (NR).

The organization of the paper is as follows. In section two, formulation of general multiobjective optimization problem as constrained single objective problem and probabilistic constraint handling is presented. In section three, probabilistic modeling for nonlinear ranking is given. In section four the probabilistic nonlinear ranking for elitism is revealed. The important implications of the probabilistic modeling are highlighted in section five. In section six a demonstrative computer experiment is given and the section is followed by discussion and conclusions.

II. OPTIMIZATION METHOD FOR MULTI-OBJECTIVITY

Weighting method is a powerful instrument for the multi-objective optimization. Its formulation in this work is adapted according to the works reported in the literature [9-11]. The weighting method deals with the weighted summation of the objective functions. Each function is associated with a weighting coefficient and weighting sum of the objectives is minimized. Thus, the multiple objective functions are expressed via a single objective function. The weighting coefficients w_i are real numbers such that $0 \le w_i$ for all objectives $i=1,\ldots,k$ so that a weighting problem can be stated as

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

In the constraint handling a single objective is used and is subject to minimization. It can be stated that

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

We assume that the form of the feasible region is given by

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

We consider that the summation of the constraint violations is another objective 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

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

Therefore, the problem definition becomes as below.

$$\min \; f(\boldsymbol{x}) + \sum_{i=1}^{m} \mu_i g_i(\boldsymbol{x}) = f(\boldsymbol{x}) + G(\boldsymbol{x}) \tag{6}$$

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

where w_1=1, $w_{2i}=\mu_i$. With this, the problem is equivalent to a single objective problem, where the objective is denoted by $f(\boldsymbol{x})$ and the constraints denoted by $g_j(\boldsymbol{x})$. The method known as *ε-Constraint* method is such an approach [11, 12]. In this method one of the objective functions is selected to be optimized, while all the other objective functions are converted into constraints. This is done by setting an upper bound to each of them. The problem to be solved is now of the form

$$\min \; f_1(\boldsymbol{x}); \text{ subject to } f_j(\boldsymbol{x}) \le \varepsilon_j \quad \text{for all j=1,2,..,k, j} \ne \text{l; x} \in \text{S} \tag{7}$$

With the above considerations we minimize $f_l(\boldsymbol{x})$; subject to $f_j(x) \le \varepsilon_j$ for all $j=1,2,\ldots,k,\ j \neq l;\ x \in S$
where $l \in \{1,\ldots,k\}$. Naturally, inequalities can be converted to equalities by taking $\varepsilon_j=0$ for all $j=1,2,\ldots,k,\ j \neq l$.
In the present case, the minimization of the function in (6) takes the form

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

where J is the number of constraints; function $g_j(\boldsymbol{x})$ is considered to be a *penalty function* and the parameters R_j are the associated *penalty parameters*. The determination of the penalty parameters is an issue and although this issue addressed in the literature [6], the issue still persists and is subject to improvements. In this work this issue is addressed by a probabilistic approach which underlies also the probabilistic sorting for effective elitism, subject matter of this work.

III. NONLINEAR RANKING WITH PROBABILISTIC CONSIDERATIONS

In general a constrained optimization (8) is written in the form

$$\min \; P(\boldsymbol{x}) = f(\boldsymbol{x}) + \sum_{i=1}^{J} \mu_j g_j(\boldsymbol{x}) \tag{9}$$

where $f(\boldsymbol{x})$ denotes 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 given by

$$\mu_j(g_j) = C\, r_j(g_j) \tag{10}$$

In (10), r_j is a new penalty parameter; C is a constant common for all constraints. As $g_j(\boldsymbol{x})$ is at each generation continually tried to be vanishing during the evolutionary minimization process, with respect to the population density of solutions, the probability density of $g_j(\boldsymbol{x})$ is highest about zero violations, and its value gradually diminishes proportional with the degree of violation. In 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, i.e. the form of the density remains the same being independent of the range it models, while the exponential pdf is a unique density having this property. With this information peculiar to the subject matter of this research, we can confidently apply the exponential pdf, which is given by

$$f_\lambda(y) = \lambda e^{-\lambda y} \tag{11}$$

where λ is the decay parameter. If we define

$$y = g_j(x) \tag{12}$$

then the pdf in (11) becomes

$$f_{g_j}(g_j) = \lambda_j e^{-\lambda_j g_j} \tag{13}$$

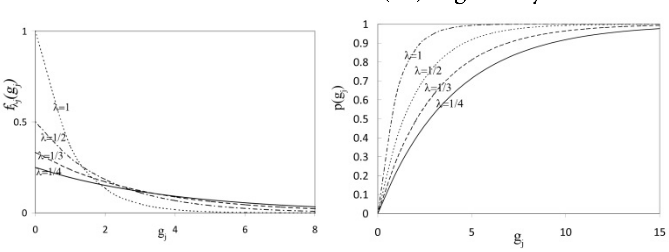
The mean value of the exponential pdf function is equal to λ_j^{-1}. During the evolutionary search $g_i(\boldsymbol{x})$ is a general form of violation, which applies to any member s of the population, and therefore, in explicit form, we can write

$$f_{g_j}(g_{j,s}) = \lambda_j e^{-\lambda_j g_{j,s}} \tag{14}$$

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

$$\lambda_j = 1/\bar{g}_j \tag{15}$$

It is to 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. The cumulative distribution function of (14) is given by



(a) (b)

Fig. 1. Plot of exponential pdf for different decay constants vs j-th violation g_j (a); $p(g_j)$ vs g_j (b)

$$p(g_j) = \frac{1}{\bar{g}_j}\int_0^{g_j} e^{-\frac{g_j}{\bar{g}_j}} dg_j = 1 - e^{-\frac{g_j}{\bar{g}_j}} \tag{16}$$

If we take $p(g_j)$ as a new random variable, the probability density $f_p(p)$ of the new random variable p is given by[13]

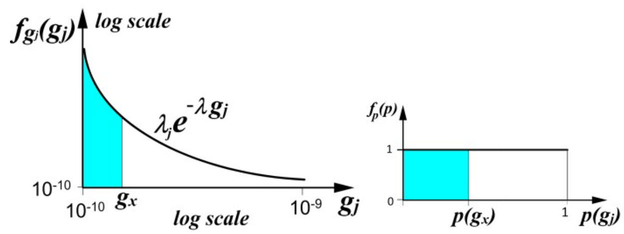
$$f_p(p) = \frac{f_{g_j}(g_j)}{\left|\frac{dH(g_j)}{dg_j}\right|_{g_j = H^{-1}(p)}} \tag{17}$$

that gives

$$f_p(p) = 1 \tag{18}$$

which is a uniform probability density shown in figure 2b together with the exponential distribution in figure 2a. In this figure the marked areas are equal having important implication in nonlinear ranking and elitism. The probability $p(g_j)$ measures the magnitude or effectiveness of a violation, so that 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 [14, 15]. Substitution of (10) into (9) yields

$$P(\boldsymbol{x}) = f(\boldsymbol{x}) + C\sum_{i=1}^{J} r_j(g_j) g_j(\boldsymbol{x}) \tag{19}$$



(a) (b)

Fig. 2. Pdf of the constraint violations in the objective functions space (a); in the probabilistic distance space (b).

where the constant C is called as *convergence parameter* as it is related to the convergence properties of the search. The new penalty parameter r_j which is a function of g_j, in general. In (19), $r_j(g_j)g_j$ is replaced by $p(g_j)$, in the form

$$r_j(g_j) g_j = p_j(g_j) \tag{20}$$

so that (19) becomes

$$P(\boldsymbol{x}) = f(\boldsymbol{x}) + C\sum_{i=1}^{J} p_j(g_j(\boldsymbol{x})) \tag{21}$$

In view of (20), r_j is given by

$$r_j = f(g_j) = p_j(g_j)/g_j \tag{22}$$

The new formulation (21) yields favourable far reaching implications which are presented in the next section.

IV. PROBABILISTIC SORTING FOR EFFECTIVE ELITISM

A. STAGE ONE: NON-DOMINATED SORTING AND ELITISM

The implementation of the theoretical results yielding an evolutionary optimization algorithm is accomplished in two stages. In the first stage non-dominated sorting ***(NS)***, tournament selection and elitism is carried out in a way essentially based on that as described in [7]. This is schematically illustrated in figure 3, where subtle details are also indicated for clarity.

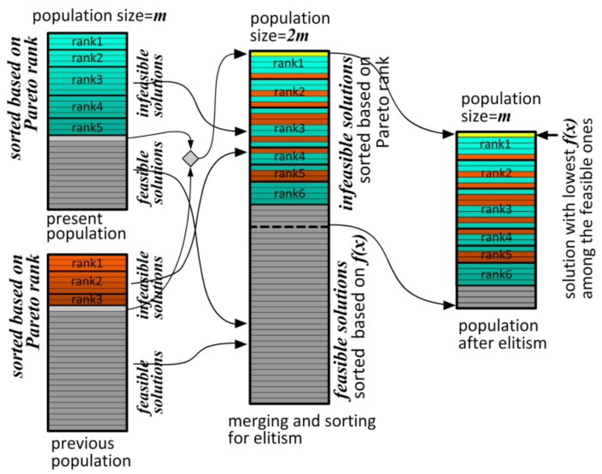


Fig. 3. NS based sorting and elitism

B. Stage two: Non-linear Ranking and Elitism

The NS algorithm described above is repeated for some number of generations so that the Pareto front sufficiently develops. Thereafter a non-linear ranking *(NR)* procedure based on the probabilistic considerations described above is employed as follows. During the tournament selection process, for two infeasible solutions from the productive domain, the value $P(g_j, \boldsymbol{x})$ in (2) is used to determine the winner of the tournament. In this procedure, clearly, a solution with lower $P(g_j, \boldsymbol{x})$ value is preferred over the solution with a larger $P(g_j, \boldsymbol{x})$ value. If a solution in the tournament belongs to the non-productive domain, then the same consequences apply as in the NS tournament. Namely, productive solutions win over non-productive solutions, and among non-productive solutions, the solution which is nearest to the productive domain wins. The possible outcomes during the non-linear ranking procedure are exemplified in figure 4. For instance, in this figure solution *B* represents the best solution among the feasible ones. When this solution is in a tournament with in infeasible solution from the productive domain, e.g. solution E, the winner of the tournament is obtained using $P(g_j, \boldsymbol{x})$. That is solution B is considered as if it were an infeasible one for

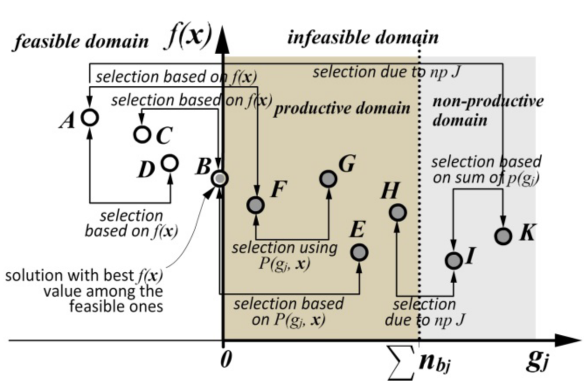


Fig. 4. Sketch for the tournament selection during NR

this comparison, so the chance *B* remains in the population is increased. For solutions from the productive domain, as $P(g_j, \boldsymbol{x})$ is a summation of function value $f(\boldsymbol{x})$ and summed up values of $p(g_j)$, population members that have a low function value and at the same time small sum of $p(g_j)$ are favored in the selection process. A solution having a low summation of $p(g_j)$ means that this solution has the unusual property that it violates several constraints with an extraordinarily low amount, when considered in perspective with the average violations of the respective constraints. In contrast to the Pareto-ranking based algorithm exercised before, the probabilistic selection mechanism will not permit solutions with low function value to remain in the population, provided the coefficient *C* is selected large enough.

The important implication of the *NR* tournament selection is assigning a commensurate right penalty parameter for every constraint, and even for each population member, where the penalty parameter is embedded in the non-linear distance function [16]. By means of this, the robustness and precision of the algorithm is guaranteed, together with the high stability of the search process. After the non-linear ranking based tournament selection, $P(g_j, \boldsymbol{x})$ is used during an elitism procedure, as seen in figure 10. From the figure it is noted that in the sorting step for the elitism the infeasible solutions are sorted based on their $P(g_j, \boldsymbol{x})$ values. Generally the mean values for the different constraints of two consecutive generations being merged for elitism differ, and it is generally expected that the mean values improve from generation to generation. In order to ensure accurate convergence, in this implementation for the sorting procedure during the NR elitism $P(g_j,\boldsymbol{x})$ is obtained using the mean value of the respective generation when the chromosome was created. This way the convergence is slowed down in order to ensure that the solutions from the past generation will also have significant influence in the ensuing generation. This is in order to maintain diversity during the search and carefully target the minimum being approached with the population. The nonlinear ranking based sorting and elitism is illustrated in figure 5.

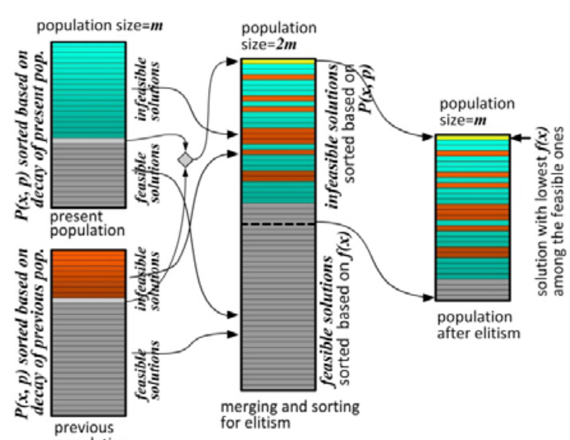
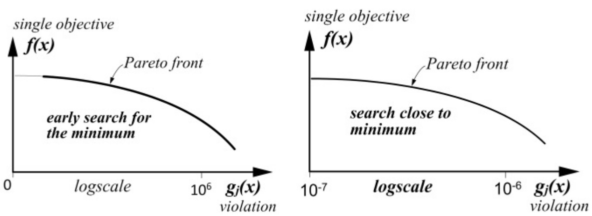


Fig. 5. NR based sorting and elitism

V. IMPLICATIONS OF THE PROBABILISTIC MODELING

A. ADAPTIVE ZOOMING FOR RANKING AND EFFECTIVE ELITISM

Adaptive zooming for ranking with precision is accomplished as follows. The favourable solutions are by accurately ranked 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 machine or genotype coding precision can allow, say at the range of 10^{-10}. A sketch of the Pareto front at the early stage of the genetic search is given in figure 6a. Illustration of the Pareto front at the last stage of the genetic search is given in figure 6b.



(a) (b)

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 2a by the indicated area. The computation of the colored area in the figure 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 coding. This situation is circumvented in figure 2b by taking simply $p(g_j)$ as the probability distance to the minimum. The indicated areas are the same and they are equal to $p(g_j)$. The indicated area in figure 2b defines 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 the transformed probabilistic space in a macro scale between zero and unity, always. This situation is equivalent to apply a commensurate mathematical 'lens' to the space formed by actual objective function and the constraint functions to carry out the convergence process without being effected by any scale of convergence happening in this space.

B. EFFECTIVE TOURNAMENT SELECTION

Two important aspects in this work, beyond the straightforward tournament selection process, are the followings.

1. In the tournament of the non-linear ranking, the present and the preceding populations is accomplished using their respective decay constants (λ). In this case the situation is depicted in figure 7, where the same rank is assigned to different violations depicted $g\lambda_{2j}$ as present violation and $g\lambda_{21j}$ as the preceding violation. By doing so, diversity in the genetic population is maintained although it slows down the convergence to some extent. However, the gain is reducing the risk of premature convergence.

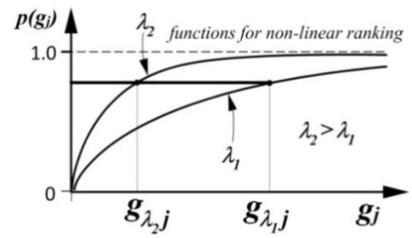


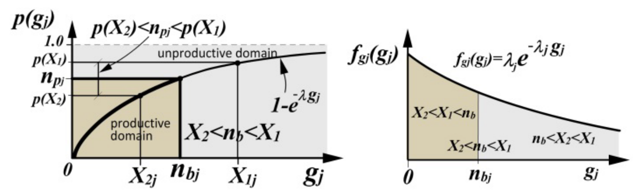
Fig. 7. Illustration of the gneration dependent ranking procedure during non-linear elitism

2. Solutions in NS as well as NR tournaments will be evaluated depending on the condition given by

$$\sum_{j=1}^{J} p(g_j) < n_{pj}\; J \qquad (23)$$

where J is equal to the number of constraints, and n_{pj} denotes a probability threshold, above which a solution is deemed *unproductive* among the infeasible solutions, and below which a solution is deemed *productive*. It has a counterpart in the objective space denoted by n_{bj}. This is seen in figure 8, where horizontal axis refers to NS (nondominated sorting) procedures and vertical axis refers to NR (nonlinear ranking) procedures.

In case one solution fulfills (23), while the other one does not, then the solution in the productive domain wins the tournament over the other one, without considering rank or



(a) (b)

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

crowding information. This case is shown in the same figure, where the violation in the productive domain is denoted by X_{2j} and its counterpart is X_{1j}. The counterpart of (23) in the objective space is given by

$$g_T = n_{b_j} \sum_{j=1}^{J} \bar{g} = \sum_{j=1}^{J} \frac{n_{b_j}}{\lambda_j} \qquad (24)$$

However, since λ_j is evolving from generation to generation, g_T is not constant. In contrast with this, in the probabilistic non-linear ranking domain, the location of maximum probability of the event that two solutions appear on either side of the threshold n_{bj} is always at n_p=0.5, irrespective of λ_j. The case for the probabilistic raking domain is illustrated in figure 9, where the variation of $p(g_j)$ with respect to n_{bj} is illustrated.

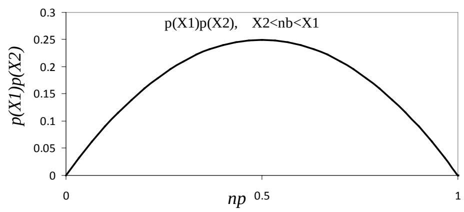


Fig. 9. Plot of the probability that two solutions occur on different sides of the thtreshold n_{pj}.

The case for the objective space is illustrated in figure 10, where the maximum occurs for $n_{bj}=ln2/\lambda_j$, which is the median of the exponential probability density shown in figure 8b. In figure 10 the single plot seen in figure 9 corresponds to a family of plots with respect to the parameter λ_j.

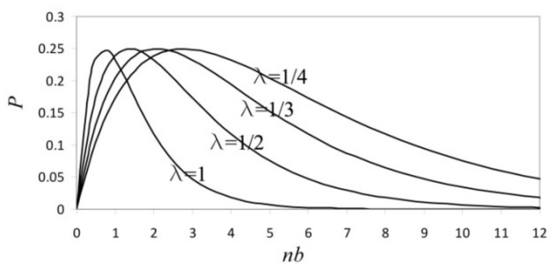


Fig. 10. Plot of the probability that two solutions occur on different sides of the threshold n_{bj}.

Explicitly, for $n_{bj}=ln2/\lambda_j$, its counterpart in terms of the probabilistic ranking domain is n_{pj}=0.5. Thus, the constant probabilistic distance measure provides an adaptive threshold for productive chromosomes throughout the generations, in any scale permitted by the machine or genotype precision. By means of this particular tournament selection procedure, the detrimental effect on 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 augmented against the contingent average increase that may occur especially during the advanced stages of the convergence. 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 described above is given by

$$P_j = P(g_j) = P(X1_j)P(X2_j) = e^{-\lambda_j n_{bj}} - e^{-2\lambda_j n_{bj}} \tag{25}$$

C. Fast and robust convergence

Thanks to the probabilistic distance providing nonlinear ranking, robust progress for convergence at each generation is obtained. To see this, from (22)

$$r_j = \frac{p(g_j)}{g_j} = \frac{1 - \mathrm{e}^{-\lambda_j g_j}}{g_j} \tag{26}$$

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

$$\lim_{g_j \to 0} \; r_j = \frac{p(g_j)}{g_j} = \lim_{g_j \to 0} \lambda_j e^{-\lambda_j g_j} = \lambda_j \tag{27}$$

The variation of the penalty parameter r_j with g_j , based on (36) is shown in figure 11.

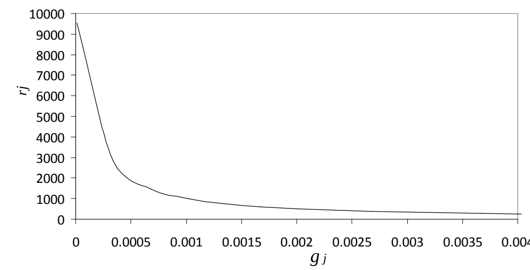


Fig. 11. Illustration of the *new* penalty parameter r as to probabilistic modeling: r=(1-exp(-λg))/g, where λ=10000

VI. COMPUTER EXPERIMENT

Computer experiments have been carried out using a standard optimization problem from the literature. The following problem is due to Floundas and Pardalos [17]. The problem consists of a single objective with 9 constraints, subject to minimization, as given by (38)-(40).

$$\textit{Minimize} \;\; f(\boldsymbol{x}) = 5\sum_{i=1}^{4} x_i - 5\sum_{i=1}^{4} x_i^{\,2} - \sum_{i=5}^{13} x_i \tag{28}$$

$$\begin{aligned}
g_1(\boldsymbol{x}) &= 2x_1 + 2x_2 + x_{10} + x_{11} - 10 \le 0 \\
g_2(\boldsymbol{x}) &= 2x_1 + 2x_2 + x_{10} + x_{12} - 10 \le 0 \\
g_3(\boldsymbol{x}) &= 2x_2 + 2x_3 + x_{11} + x_{12} - 10 \le 0 \\
g_4(\boldsymbol{x}) &= -8x_1 + x_{10} \le 0 \\
g_5(\boldsymbol{x}) &= -8x_2 + x_{11} \le 0 \\
g_6(\boldsymbol{x}) &= -8x_3 + x_{12} \le 0 \\
g_7(\boldsymbol{x}) &= -2x_4 - x_5 + x_{10} \le 0 \\
g_8(\boldsymbol{x}) &= -2x_6 - x_7 + x_{11} \le 0 \\
g_9(\boldsymbol{x}) &= -2x_8 - x_9 + x_{12} \le 0
\end{aligned} \tag{29}$$

where the ranges for the independent variables are given by

$$0 \le x_i \le 1 \;\; (i = 1,...,9); 0 \le x_i \le 100 \;\; (i = 10,11,12); 0 \le x_i \le 1 \;\; (i = 1 \tag{30}$$

The best known optimum is

$f(\boldsymbol{x}^*)$=-15.0,

and the corresponding best variable values are $\boldsymbol{x}^*$=(1,1,1,1,1,1,1,1,1,3,3,3,1).

The algorithm is executed with the following settings: population size=200; amount of generations=100; C=1000; ratio of NS/NR procedures=4/1; crossover probability=0.9; mutation probability=0.05. The results are shown in figure 12-15 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 4 Pareto-ranking based generations, followed by one probabilistic selection based generation.

After 10 iterations the best feasible solution is found to be *f(**x**)*=-13.98583864.
The population is shown in figure 12.

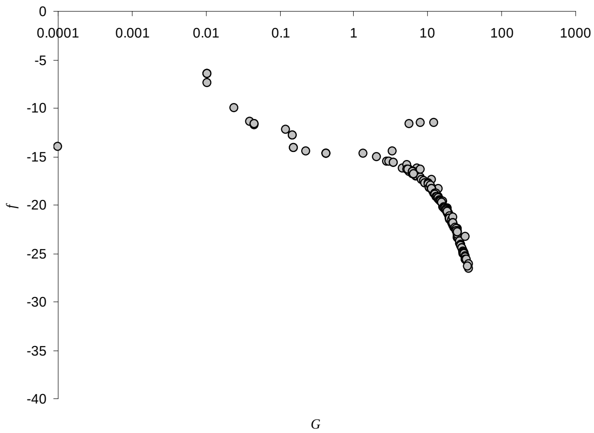


Fig. 12. Population after the 10th iteration; horizontal axis is the total violation *G* on a log scale; vertical axis is *f(**x**)*

After 20 iterations the best feasible solution is found to be *f(**x**)*=-14.9076345785146.

The population is shown in figure 12. The figures 12-17 demonstrate the robust convergence properties of the algorithm. Namely, the population members form a compact aggregation about the close vicinity of the optimum. This aggregation makes the mean of the violations g_j small, so that the decay constant λ_j of the exponential pdf becomes large, and consequently the slope of the penalty parameter *r* is large. Due to this, the convergence to the optimum is fast, accurate, and with precision. Due to the memoryless ness property of the exponential pdf, the populations form about the same patterns in any scale of the convergence process. This is clearly seen in the figures by the logarithmic scale of representations of the violations.

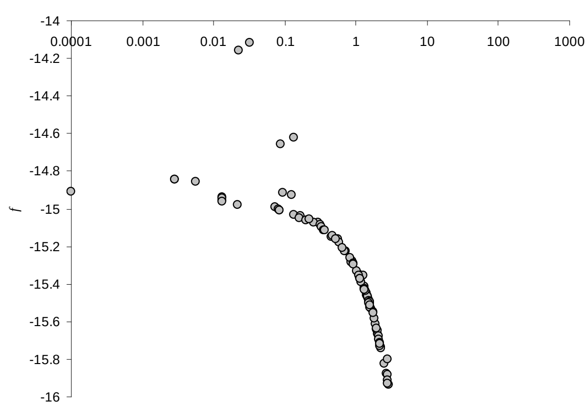


Fig. 13. Population after the 20th iteration; horizontal axis is the total violation *G* on a log scale; vertical axis is *f(**x**)*

After 30 iterations the best feasible solution is found to be *f(**x**)*=-14.9760230713287.
The population is shown in figure 14.

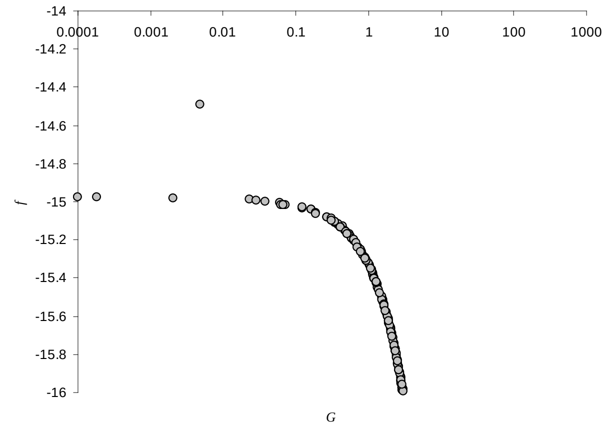


Fig. 14. Population after the 30th iteration; horizontal axis is the total violation *G* on a log scale; vertical axis is *f(**x**)*

After 50 iterations the best feasible solution is found to be *f(**x**)*=-14.9985221605613.

The population is shown in figure 14. The independent variable values of this solution are x_1=0.999997312719596; x_2=0.999997982197311; x_3=0.999999888524811; x_4=0.999999871166525; x_5=0.999994649877324; x_6=0.999987862005421; x_7=0.999984815877352; x_8=0.999999999750139; x_9=0.999926599531956; x_{10}=2.99995794671011; x_{11}=2.99961604207534; x_{12}=2.99907993443006; x_{13}=0.999999037205755.

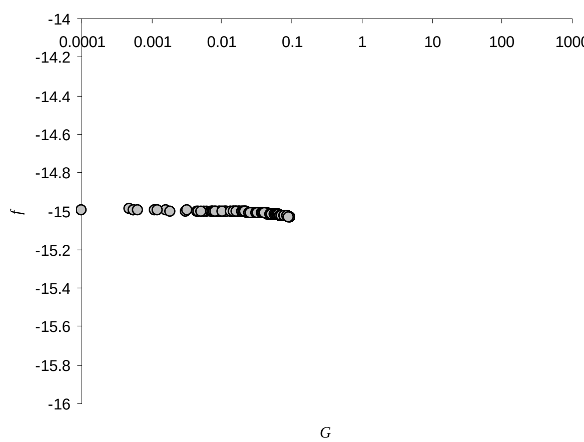


Fig. 15. Population after the 50th iteration; horizontal axis is the total violation *G* on a log scale; vertical axis is *f(**x**)*

After 80 iterations the best feasible solution is found to be *f(**x**)*=-14.999997075874.

The whole population is shown in figure 12. The independent variable values of this solution are x_1=0.999999970028148; x_2=1; x_3=0.999999993220535;

x_4=0.999999971015986; x_5=0.999999868494369; x_6=0.999999960482284; x_7=0.999999981750632; x_8=0.999999947363726; x_9=0.999999932369763; x_{10}=2.99999980083455; x_{11}=2.99999890685553; x_{12}=2.99999900736034; x_{13}=0.999999999039419.

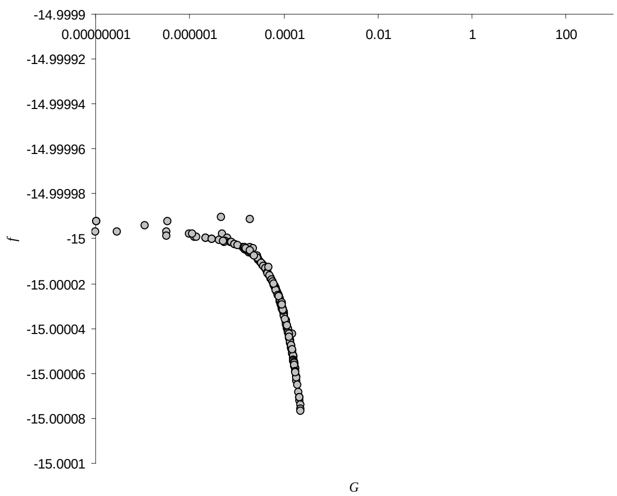


Fig. 16. Population after the 80th iteration; horizontal axis is the total violation *G* on a log scale; vertical axis is $f(\boldsymbol{x})$

After 100 iterations the best feasible solution is found to be $f(\boldsymbol{x})$=-14.9999999458368.

The whole population is shown in figure 15. The independent variable values of this solution are x_1=0.999999999790223; x_2=0.999999999809861; x_3=0.99999999933798; x_4=0.99999999995506; x_5=0.999999994377998; x_6=0.999999995023679; x_7=0.999999999831045; x_8=1; x_9=0.999999996761354; x_{10}=2.99999999225267; x_{11}=2.99999998081316; x_{12}=2.99999999477602; x_{13}=0.999999997535212.

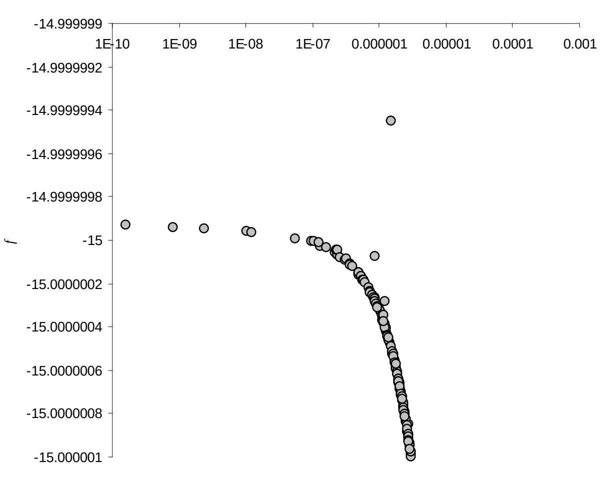


Fig. 17. Population after the 100th iteration; horizontal axis is the total violation *G* on a log scale; vertical axis is $f(\boldsymbol{x})$

VII. CONCLUSIONS

Probabilistic sorting for effective elitism in multi-objective evolutionary algorithms is presented. In the evolutionary optimization ranking of the genetic population members plays very important role on the performance of the algorithm. This work addresses this issue by a new non-linear ranking procedure, which eventually leads to an effective elitism and marked performance of the algorithm. Conventionally, in constrained or multi-objective optimization problems evolutionary computation turns out to be supported by auxiliary optimization means, in order to approach the optimum sufficiently close. In this respect, by means of the new methodology a marked improvement is achieved. The source of the improvement lies in the non-linearity of the ranking, achieved by the transformation of the objective space to a newly defined probabilistic distance domain. The transformation is adaptively carried out throughout the generations, so that the commensurate ranking with respect to the generation is maintained. Additionally, explicit definition of productive and non-productive chromosomes has been made, and accordingly maximum gain from the unproductive chromosomes is exported to the productive portion of the population at each generation. By means of the particular tournament selection procedure, the detrimental effect on the average violation by the stiff constraints, that is, by highly non-productive population members is prevented; namely, during two consecutive generations the progressive diminishing of the average is augmented against the contingent average increase that may be effective especially during the advanced stages of the convergence. Non-linear ranking plays two major roles at the same time. One is the accomplishment of an adaptive penalty parameter matching the optimality conditions during the search. The other is maintaining maximum gain constantly from unproductive to productive solutions. This allows accurate and systematic convergence with precision, which is also rapid. The probabilistic sorting is implemented in both, nonlinear tournament selection and elitism. The method showed outstanding performance as to speed of convergence, precision and approaches to the solution without auxiliary support like local search, memetic algorithm etc. This is exemplified by means of a standard problem chosen from the literature for the comparison of the results and demonstration of the effectiveness of the methodology. 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 results with the theoretical underlying material. It is also noteworthy to mention that, due to the systematic convergence procedure established by the novel method, the search process is demonstrated to be transparent.

REFERENCES

[1] C. M. Fonseca, "An overview of evolutionary algorithms in multiobjective optimization," *Evolutionary Computation,* vol. 3, pp. 1-16, 1995.
[2] C. A. C. Coello, "An updated survey of Ga-based multi-objective optimization techniques," *ACM Computing Surveys,* vol. 32, pp. 109-143, 2000.
[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] K. Deb and R. Datta, "A bi-objective constrained optimization algorithm using a hybrid evolutionary and penalty function approach," *Engineering Optimization,* vol. 45, pp. 503-527, 2013.
[7] 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.
[8] C. A. C. Coello, "Use of a self-adaptive penalty approach for engineering optimization problems," *Ciomputers in Industry,* vol. 41, pp. 113-127, 2000.
[9] S. Gass and T. Saaty, "The computational algorithm for the parametric objective function," *Naval Research Logistics Quarterly,* vol. 2, p. 7, 1955.
[10] L. Zadeh, "Non-scalar-valued performance criteria," *IEEE Trans. Automatic Control,* vol. 8, p. 2, 1963.
[11] K. Miettinen, *Nonlinear Multiobjective Optimization*. Boston: Kluwer Academic, 1999.
[12] 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.
[13] P. L. Meyer, *Introductory Probability and Statistical Applications*: Addison-Wesley, 1970.
[14] G. Bachman and L. Narici, *Functional Analysis*. New York: Dover, 2000.
[15] J. T. Oden and L. F. Demkowicz, *Applied Functional Analysis*: CRC Press, 1996.
[16] O. Ciftcioglu, M. S. Bittermann, and I. S. Sariyildiz, "Precision Evolutionary Optimization - Part I: Nonlinear Ranking Approach," presented at the GECCO 2012, Philadelphia, 2012.
[17] C. Floundas and P. Pardalos, *A collection of test problems for constrained global optimization, Vol.455*. Berlin, Germany: Springer Verlag, 1987.

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.

Michael S. Bittermann was born in Germany in 1976. He graduated with the highest honors as MSc in Architecture from Delft University of Technology (TU Delft), The Netherlands, in 2003. He received his PhD degree in Design Informatics with the highest honors from the same university in 2009. Bittermann was distinguished as one of the five eminent researchers among all 2008-2009 PhD graduates of TU Delft in the form of the Young Researcher Fellowship of Delft University of Technology (80k€). He joined Maltepe University, Istanbul, Turkey in 2013, where he contributed to the research project titled 'Design by Cognition and Comprehension,' funded by the Scientific and Technological Research Council of Turkey. His research interest is computational cognition.

Ramazan Caglar received the BSc., and MSc. diploma in Electrical Engineering from the Istanbul Technical University in 1984 and 1987 respectively, and PhD degrees in Graduate School of Science, Engineering and Technology of the same university in 1999. He accepted as Postdoctoral Researcher by the School of Electric and Computer Engineering at Georgia Institute of Technology, Atlanta, USA in 2001. He has conducted his research activities for two years in this university. His research interests are power system modeling, reliability of electric power system, the effects of deregulation on the electric power system, power transmission pricing, distributed generations, and fault diagnosis, risk management and signal processing.

In 1984 he joined the Faculty of Electrical Engineering, Istanbul Technical University as a research assistant, where he is presently an Assoc. Professor. He is active in teaching and research in the general modeling, analysis, and control of power systems. His has published article in some referred journals, and conference proceedings.

Dr. Caglar is a member of IEEE, IEEE-PES, a member of CIGRE, a member of Turkish National Committee on Illumination (ATMK), and a National Chamber of Turkish Electrical Engineering (EMO).

Rituparna Datta received the PhD degree from the Indian Institute of Technology Kanpur (IITK), Kanpur, India, in 2013.

He is a Post-doctoral researcher with Graduate School of Knowledge Service Engineering, Department of Industrial & Systems Engineering at Korea Advanced Institute of Science and Technology (KAIST). Prior to that he was a Post-doctoral fellow in French Institute for Research in Computer Science and Automation (Institut national de recherche en informatique et en automatique, INRIA), France, Project scientist in Smart Materials, Structures and Systems Laboratory, Indian Institute of Technology, Kanpur, India and Postdoctoral research fellow with the Robot Intelligence Technology (RIT) Laboratory at KAIST. His current research interests include investigation of evolutionary computation, constraint handling, multiobjective optimization, knowledge extraction from data, and robotics. His has two edited books with Springer, many international journals and international conferences in his credit.

Dr. Datta is a regular Reviewer of the IEEE Transactions on Evolutionary Computation, IEEE Transactions on Cybernetics, The Journal of Applied Soft Computing, and The Journal of Engineering Optimization.

THE JOURNAL OF
COGNITIVE SYSTEMS
VOLUME 01 NUMBER 01 DECEMBER 2016

PROBABILISTIC CONSIDERATIONS UNDERLYING A NOVEL EVOLUTIONARY COMPUTATION

Tahir Cetin Akinci[1], Ozer Ciftcioglu[2], Jelena Dikun[3], Emine Ayaz[1]

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

Penalty function approach used for optimization has a growing interest in the literature due to its effectiveness not only for multiobjective optimization but also for constraint optimization. Although there are several excellent papers on the penalty function approaches, up till now there is no clear method for the systematic selection of penalty parameters per constraint since the topic is quite elusive. The issues being well realized, there are several researches addressing these issues to some extent. However, still the robustness of these methods remains the main issue due to some newly added additional parameters subject to determination. This work endeavors to address this issue and first it makes a systematic analysis. Following the analysis it establishes a probabilistic approach as the issue is entirely in the domain of probability. According to the best knowledge of the authors the approach is unique as to probabilistic treatment of the issue. The approach models the probability density of the random population throughout the generations and based on this, penalty parameters are determined following the probabilistic derivations. The theoretical considerations are substantiated by computer experiments and a demonstrative example is presented showing the salient effectiveness of the approach..

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

I. INTRODUCTION

EVOLUTIONARY multiobjective optimization is a popular approach in science and engineering. It is particularly important in cognitive science, because of the decision-making process and ensuing optimization process for action and communication. In this work, constraint optimization is the subject matter, which is to consider as multiobjective optimization due to the method of Penalty function approach. Its use for constraint optimization has a growing interest in the literature, due to its effectiveness not only for multiobjective optimization but also for constraint optimization. Although there are several excellent papers on the penalty function approaches, up till now there is no a clear method for the systematic selection of penalty parameters per constraint, since the topic is quite elusive. The issues of common penalty parameter pertinent to all constraints are well understood. Still the robustness of these methods remains the main issue due to variation of the parameters during the optimization process. The penalty function methods are widely used methods for evolutionary constraint optimization, which differ from each other due to some different strategies. In this respect some examples are static penalty, dynamic penalty, annealing penalty, adaptive penalty, co-evolutionary penalty, death penalty and their associated penalty parameters [1-10]. Strategies that did not require a penalty parameter were proposed in the literature, e.g. [11, 12], while the latter work was later superseded by the penalty function approach [13]. This variety of penalty-function oriented researches is the manifestation of the persisting issue of determining the penalty parameters with respect to each constraint.

In this work a new approach is proposed. Probabilistic considerations underlying the approach are described in detail. The approach is based on the evolutionary probabilistic modeling of the random solutions and the introduction of a probabilistic distance metric. The model is used for effective ranking of genetic population members and thereby yields efficient converging solutions.

The organization of the paper is as follows. In section two, formulation of the general multiobjective optimization problem and non-linear ranking are presented. In section three, important implications of the evolutionary probabilistic approach are described. In section four a demonstrative computer experiment is given. The section is followed by conclusions.

II. PROBLEM FORMULATION AND NON-LINEAR RANKING

A. WEIGHTING METHOD

A well-defined method for dealing with the multi-objective optimization is known as *weighting method* [14-16]. In this method each objective is associated with a weighting coefficient and minimizes the weighting sum of the objectives. In this way, the multiple objective functions are transformed

Manuscript received October 10, 2016; accepted November 24, 2016.
Corresponding author: Özer Ciftcioglu (E-mail: o.ciftcioglu@tudelft.nl).
Digital Object Identifier:

ISSN 2548-0650 © 2016 The Journal of Cognitive Systems (JCS). Personal use is permitted, but republication/redistribution requires JCS permission. See www.dergipark.gov.tr/jcs for more information.

into a single objective function. We assume that the weighting coefficients w_i are real numbers such that $0 \le w_i$ for all objectives i=1,….,k so that a weighting problem can be stated as

$$\min \sum_{i=1}^{k} w_i f_i(\boldsymbol{x}); \;\; \boldsymbol{x} \in S \qquad (1)$$

In the constraint handling in this work a single objective is involved which is subject to minimization. Therefore the problem can be stated as

$$\begin{aligned} &\min \quad f(\boldsymbol{x}) \\ &\textit{subject to} \;\; g(\boldsymbol{x}) = [g_1(x), g_2(x), ..., g_m(x)]^T \le 0 \end{aligned} \qquad (2)$$

We assume that the feasible region is of the form

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

If we assume 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

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

where

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

Hence, the problem definition becomes

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

With this formulation, the weighting method becomes appropriate to employ where $w_1=1$, $w_{2i}=\mu_i$. In (6) the problem formulation becomes two objective functions subject to minimization or alternatively a single objective function with an objective vector subject to minimization. This formulation of the problem is equivalent to a single objective problem with constraints where the constraints are given by the vector $g(\boldsymbol{x})$ which is considered to be a penalty function and the parameters μ_i are the associated penalty parameters. We formulate the multiobjective optimization as two-objective optimization, which can be further treated as single objective optimization with constraints, without deviating from generality. This approach is known to be as *ε-Constraint* method [16, 17]. Among the objective functions one function is selected to be optimized, and, by setting an upper bound to each of them, all the other objective functions are converted into constraints. The problem now has the form

minimize $f_l(x)$

subject to $f_j(x) \le \varepsilon_j$ for all $j=1,2,….,k,\; j \ne l;\; x \in S;\; l \in \{1,…,k\}$

The inequalities can be transformed to equalities by considering $\varepsilon_j=0$ for all $j=1,2,….,k,\; j \ne l$. Based on the above considerations, we assume the problem formulation as a constraint optimization with single objective, so that in a general constrained optimization problem the problem formulation is written as

$$P(\boldsymbol{x}) = f(\boldsymbol{x}) + \sum_{i=1}^{J} \mu_j g_j(\boldsymbol{x}) \qquad (7)$$

In (7), we make the following observation. Since in the weighting method the weights are positive, in (7) the penalty parameter μ_j is positive. This implies that in a general case we can surmise that the optimization problem is in the form as depicted in figure 1.

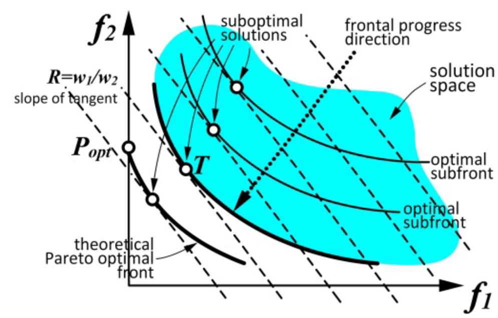


Fig. 1. Approach to the final optimal solution by means of constant penalty parameter R.

Referring to (6) in figure 1 $f_2(\boldsymbol{x})=f(\boldsymbol{x})$ and $f_1(x)=g(\boldsymbol{x})$ denoting violations; also in place of multiple μ_i each of which belong to one constraint, we can consider both a common and constant penalty parameter R which is the slope of the tangent of the Pareto optimal front during the progressive search of the front. In figure 1, the slope of the tangent being negative, the violations are represented as negative quantities so that μ_j and $g_j(x)$ become positive quantities. If we consider that the optimal front is a series of solutions determined by the tangent points of the tangent line and the optimal front, we conclude that the optimal front is simply the envelope of the tangents. This envelope is established as follows.

We assume that a theoretical optimal front compromises the solutions between the objectives $f(\boldsymbol{x})$ and $g(\boldsymbol{x})$ where objective $g(\boldsymbol{x})$ admits to be minimally zero. In this case each solution on the optimal front can individually be represented by a line that is tangent to the optimal front at that particular solution. The parametric representation of the tangent is given by

$$\frac{f(\boldsymbol{x})}{t} + \frac{g(\boldsymbol{x})}{P_{opt} - t} = 1 \qquad (8)$$

where t is the parameter. In (8), P_{opt} is the optimum solution located at the point $f(\boldsymbol{x}) = P_{opt}$ and $g(\boldsymbol{x}) = 0$. From (8), we write

$$f(\boldsymbol{x}) = \frac{t}{t - P_{opt}(\boldsymbol{x})} g(\boldsymbol{x}) + t \qquad (9)$$

where the slope of the tangent is given by

$$r = \frac{t}{t - P_{opt}(\boldsymbol{x})} \qquad (10)$$

as a *new* penalty parameter r. The envelope of the tangent is shown in figure 2. The Pareto front is obtained by arranging (10) with respect to t and admitting a single solution for it; namely,

$$t^2 + [g(x) - f(x) - P_{opt}(x)]t + f(x)P_{opt}(x) = 0 \quad (11)$$

$$[f_1(\boldsymbol{x}) - f_2(\boldsymbol{x}) - P_{opt}(\boldsymbol{x})]^2 - 4 f_2(\boldsymbol{x}) P_{opt}(\boldsymbol{x}) = 0 \quad (12)$$

then the optimal front is obtained by equating the discriminant to zero that gives the envelope of the tangent as the optimal front. The new penalty parameter *r* is zero for *t=0* and it monotonically increases as *t* increases. For $t=P_{opt}$ the penalty parameter *r* goes to infinity.

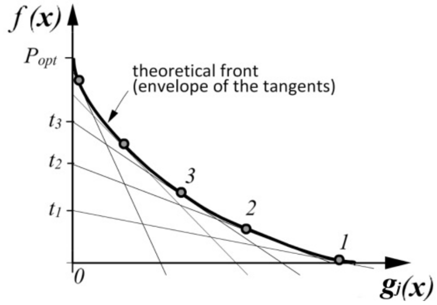


Fig. 2. The envelope of tangent and the *new* penalty parameter *r*. $r=(P_{opt}-T)/T$ where $T=P_{opt}-t$

If we consider the optimal front for each constraint separately, (10) can be written as

$$r_j = \frac{t}{t - P_{opt}(\boldsymbol{x})} \quad (13)$$

so that (7) becomes

$$P(\boldsymbol{x}) = f(\boldsymbol{x}) + \sum_{i=1}^{J} r_j g_j(\boldsymbol{x}) \quad (14)$$

B. Probabilistic Distance Metric

In (14) $g_j(\boldsymbol{x})$ at each generation continually is tried for its vanishing during the evolutionary optimization process. This is accomplished by the evolutionary algorithm, giving higher probability of reproduction to population members with small g_j values. Therefore, with respect to the population density of solutions, the probability density of $g_j(\boldsymbol{x})$ is highest about zero violations, and the density 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 (GA). 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, i.e. the form of the density remains the same being independent of the range it models and exponential pdf is a unique density having this property. With this information peculiar to the subject matter of this research, we can confidently apply the exponential probability density function (pdf), which is given by

$$f_\lambda(y) = \lambda e^{-\lambda y} \quad (15)$$

where λ is the decay parameter. If we define

$$y = g_j(x) \quad (16)$$

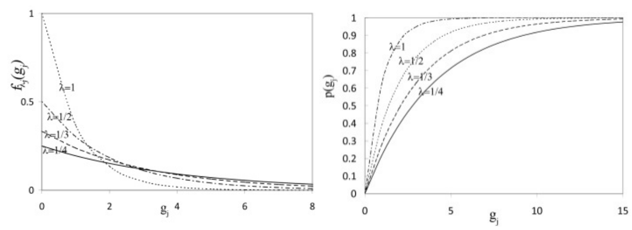
the pdf in (15) becomes

$$f_{g_j}(g_j) = \lambda_j e^{-\lambda_j g_j} \quad (17)$$

The mean value of the exponential pdf function is equal to λ_j^{-1}. During the evolutionary search $g_j(\boldsymbol{x})$ 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

$$f_{g_j}(g_{j,s}) = \lambda_j e^{-\lambda_j g_{j,s}} \quad (18)$$

The variation of the exponential pdf for different decay parameters is shown in figure 3a.



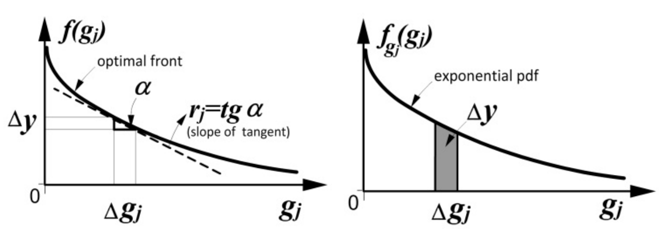
(a) (b)

Fig. 3. Plot of exponential pdf for different decay constants vs *j*-th violation g_j (a); Plot of $p(g_j)$ vs g_j for various mean values of $p(g_j)$ (b)

The mean value of the violations g_j is the characteristic of the constraint *j* and it defines the shape of the exponential distribution of the violations representing the decay constant

$$\lambda_j = 1/\bar{g}_j \quad (19)$$

The typical shape of the optimal front in figure 4, and the variation of the exponential distribution is shown together in figure 4, where 4a indicates the optimal front and 3b indicates the exponential probability density.



(a) (b)

Fig. 4. Forming the optimal front as an envelope of a slope by means of the probilistic modelling of rnadom solutions as exponential distribution.

In figure 4b, a small change in violation g_j causes small change in probability density and the probability of violations in this interval is given by

$$\Delta y_f = f_{g_j}(g_j)\Delta g_j \quad (20)$$

From figure 4a we note that a small change in the violation g_j causes a small change in the objective function along the optimal front, and it is given by

$$\Delta y_p = r_j(g_j)\Delta g_j \tag{21}$$

During the search evolution, at each generation the decay constant is newly estimated by the mean of the violations as given by(19), so that λ_j is assumed to be constant from one generation to another. Hence, (17) becomes

$$f_{g_j}(g_j) = \frac{1}{\bar{g}_j} e^{-g_j/\bar{g}_j} \tag{22}$$

In the same way, at each generation r_j is newly estimated, so that r_j is assumed to be constant from one generation to another. Taking infinitesimally small violation intervals, and equating (20) and (21), that is, equating to the objective function change to the probability in the interval dg_j we write

$$r_j(\bar{g}_j)dg_j = \frac{1}{\bar{g}_j} e^{-g_j/\bar{g}_j} dg_j \tag{23}$$

It is to note that by means of this equation above we are relating the objective function space to a probability domain in the form of a transformation. The important implications of this transformation are presented in section 3.

Defining

$$p(g_j) = \frac{1}{g_j}\int_0^{g_j} e^{-\lambda_j g_j} dg_j = 1 - e^{-\lambda_j g_j} \tag{24}$$

Integration of (24) from zero to g_j gives

$$\int_0^{g_j} r_j(\bar{g}_j)dg_j = \int_0^{g_j} \frac{1}{\bar{g}_j} e^{-g_j/\bar{g}_j} dg_j \tag{25}$$

That is,

$$r_j(\bar{g}_j)g_j = 1 - e^{-g_j/\bar{g}_j} = p(g_j) \quad , \tag{26}$$

or briefly

$$r_j g_j = p(g_j)\,. \tag{27}$$

The variation of $p(g_j)$ with g_j is shown in figure 4b.

In (7), μ_j is replaced by Cr_jg_j, namely

$$\mu_j = Cr_j g_j \tag{28}$$

where C is a constant, and the substitution of (28) into (7) with the consideration of (27) yields

$$P(\boldsymbol{x}) = f(\boldsymbol{x}) + C\sum_{i=1}^{J} p(g_j) \tag{29}$$

where J denotes the number of constraints; C is a common constant for all constraints. The probability $p(g_j)$ controls the penalty parameter $\mu_i(g_j)$ in (7), which is absorbed in $p(g_j)$. The importance of this transformation, namely from $\mu_i g_j$ to $p(g_j)$ is mainly due to its use for ranking in the probabilistic domain during the genetic optimization process.

In view of (27), r_j is given by

$$r_j = f(g_j) = p_j(g_j)/g_j \tag{30}$$

The variation of the slope r_j versus g_j is plotted in figure 5, where the variation of the slope given by (13) is also plotted.

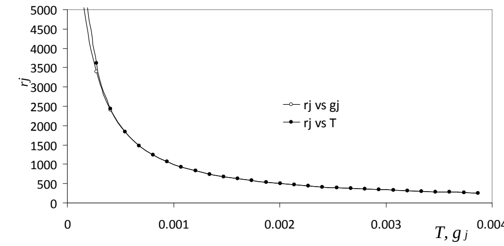


Fig. 5. Illustration of the *new* penalty parameter r as to probabilistic modeling: $r=(1-\exp(-\lambda g))/g$ and as to bi-objective formulation: $r=t/(P_{opt}-t)$

The two slopes, namely one obtained as the tangent, the envelope of which forms the Pareto front, and the other one obtained from a probabilistic model, introduced in this research, coincide satisfactorily, as seen in the figure.

The probability $p(g_j)$ is a *probabilistic distance function* or a *metric* measuring the distance from the zero violation, as it fulfils all the conditions to be a distance measure [18, 19]. The probability density of this distance metric given by (26) is computed by

$$f_p(p) = \frac{f_{g_j}(g_j)}{\left|\frac{dp(g_j)}{dg_j}\right|_{g_j=p^{-1}(p)}} \tag{31}$$

which gives

$$f_p(p) = 1 \tag{32}$$

as uniform pdf. The defined distance metric in the probability domain $p(g_j)$ is used for ranking the chromosomes for effective tournament selection and elitism, in place of remaining in the objective function space. The important implications of this transformation from objective function space to the probability domain are given in the next section.

III. Implications of the Probabilistic Distance Metric

A. Stiffness Handling

The stiffness is defined as the large numerical difference among several constraints subject to minimization. If there is stiffness among the constraints, the summation in (6) is dominated by the constraints, the pdfs of which have small decay constants. However, by using the probabilistic distance measure varying between zero and unity, this drawback is eliminated. The treatment is illustrated in figure 6, where the probabilistic distances for the constraints random variables $g_{\lambda 1j}$ and $g_{\lambda 2j}$ are the same, and the distance is between zero and unity. By giving the same priority or rank in the tournament selection process for the constraints $g_{\lambda 1j}$ and $g_{\lambda 2j}$ we consider purely their associated probabilities based on the probabilistic model without imposing any bias about the nature of the constraints.

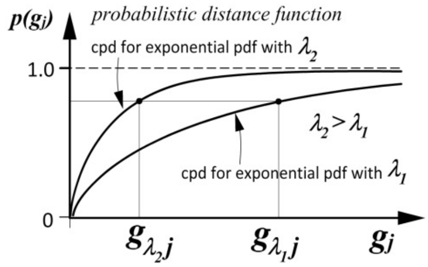


Fig. 6. Illustration of the stiffness handling

B. IMPARTIAL ELITISM SELECTION

During elitism we consider the population from the preceding generation. Therefore, below first we compute the probability of having smaller constraint violation. If we consider two exponential probability density functions with the random variables X_1 and X_2 and the associated decay parameters λ_1 and λ_2 respectively, the probability $P(X_2<X_1)$ is computed as follows. The probability of $g_2 \le X_1$, namely $P(g_2 \le X_1)$ is given by

$$P(g_2 \le X_1) = \int_{g_2}^{\infty} \lambda_1 e^{-\lambda_1 g_1}\, dg_1 = e^{-\lambda_1 g_2} \tag{33}$$

so that

$$\begin{aligned} P(X_2 < X_1) &= P(g_2 \le X_1)\, P(g_2 \le X_2) \\ &= \int_0^{\infty} e^{-\lambda_1 g_2} \lambda_2 e^{-\lambda_2 g_2}\, dg_2 = \lambda_2 \int_0^{\infty} e^{-(\lambda_1+\lambda_2) g_2}\, dg_2 \end{aligned} \tag{34}$$

which gives

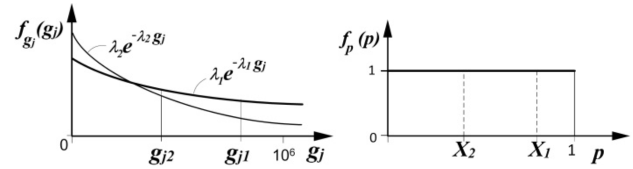
$$P(X_2 < X_1) = \frac{\lambda_2}{\lambda_1 + \lambda_2} \tag{35}$$

Let us carry out the same calculations with respect to the random variable P, the pdf of which is given by (32).

$$P(g_2 \le X_1) = \int_{g_2}^{1} dg_1 = 1 - g_2 \tag{36}$$

$$\begin{aligned} P(X_2 < X_1) &= P(g_2 \le X_1)\, P(g_2 \le X_2) \\ &= \int_0^{1} (1 - g_2)\, dg_2 = \frac{1}{2} \end{aligned} \tag{37}$$

This result shows that, irrespective to the decay constants λ_1 and λ_2 of two exponential *pdf*s, the probability $P(X_2 \le X_1)$ is always 0.5, which means ranking the random solutions during the genetic search, probabilistic distance function $p(g_j)$ is fully impartial with respect to the decay constants. This means, although the decay constants vary and they are updated each iteration, this is not reflected to the elitism. With other words, the procedure is not biased by apparently less favorable population of the latest generation due to higher probabilistic distances caused by higher decay constant. Instead, the preceding generation and the following generation are treated in perspective without bias, eliminating the decay constant factor in the computation. The exponential pdf $f_{gj}(g_j)$ in (17) and uniform pdf in (32) are sketched in figures 7a and 7b, where the random variables X_1, X_2 and two corresponding violations g_{j1}, g_{j2} are also shown. It should be pointed out that the uniformity of the uniform pdf is not affected even if the modelling error in the surmised exponential pdf model exists.



(a) (b)

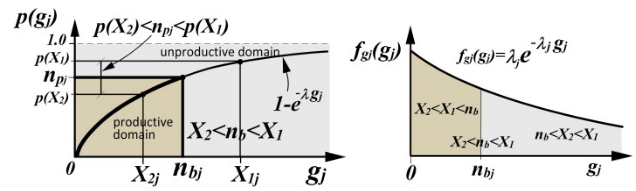
Fig. 7. Probability density $f_{gj}(g_j)$ vs g_j (a) Probability density $f_P(p)$ vs p (b)

Two important aspects in this work, beyond the basic elitism procedure, are the followings.

1. During the elitism, the combination of the present and the preceding populations is accomplished using their respective decay constants (λ). In this case the situation is depicted in figure 6, where the same rank is assigned to different violations depicted $g\lambda_{2j}$ as present violation and $g\lambda_{21j}$ as the preceding violation. By doing so, diversity in the genetic population is maintained although it slows down the convergence to some extent. However, the gain is reducing the risk of premature convergence.
2. Solutions during tournament selection will be evaluated depending on the condition given by

$$\sum_{j=1}^{J} p(g_j) < n_{pj}\, J \tag{38}$$

where J is equal to the number of constraints, and n_{pj} denotes a probability threshold, above which a solution is deemed *unproductive* among the infeasible solutions, and below which a solution is deemed *productive*. It has a counterpart in the objective space denoted by n_{bj}.



(a) (b)

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

In case one solution fulfills (38), while the other one does not, then the solution in the productive domain wins the tournament over the other one, without considering rank or crowding information. This case is shown in the same figure, where the violation in the productive domain is denoted by X_{2j} and its counterpart is X_{1j}. The counterpart of (38) in the objective space, and is given by

$$\bar{g}_T = n_{b_j} \sum_{j=1}^{J} \bar{g} = \sum_{j=1}^{J} \frac{n_{b_j}}{\lambda_j} \tag{39}$$

Referring to figure 8b, the probability P_j of the event relevant to the case described above is given by $X_2<nb<X1$, and

$$P_j = P(g_j) = P(X1_j)P(X2_j) = e^{-\lambda_j n_{bj}} - e^{-2\lambda_j n_{bj}} \tag{40}$$

However, since λ_j is evolving from generation to generation, g_T is not constant. In contrast with this, in the probabilistic non-linear ranking domain, the location of maximum probability of the event that two solutions appear on either side of the threshold n_{bj} is always at n_p=0.5, irrespective of λ_j. The case for the probabilistic raking domain is illustrated in figure 9, where the variation of $p(g_j)$ with respect to n_{bj} is illustrated also for the productive and unproductive domains.

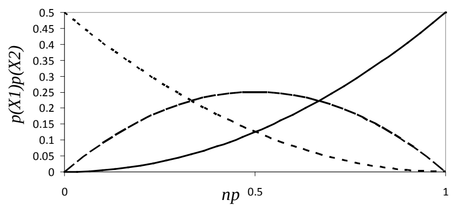


Fig. 9. Plot of the probabilities for different conditions that can arise during binary tournament selection

The case for the objective space is illustrated in figure 10, where the maximum occurs for $n_{bj}=ln2/\lambda_j$, which is the median of the exponential probability density shown in figure 8b. The single plot for each of the three possible conditions, during a binary tournament seen in figure 9, depending on the occurrence of solutions in productive or non-productive domains, correspond to a family of plots with respect to the parameter λ_j in figures 10-11.

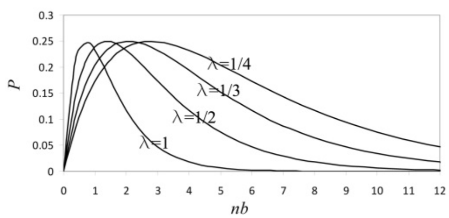


Fig. 10. Plot of the probability that two solutions occur on different sides of the threshold n_{bj} for λ=1, 1/2, 1/3, 1/4. The respective maximum occurs at n_b=0.693/λ

Explicitly, for $n_{bj}=ln2/\lambda_j$, its counterpart in terms of the probabilistic ranking domain is n_{pj}=0.5. Thus, the constant probabilistic distance measure provides an adaptive threshold for productive chromosomes throughout the generations, in any scale permitted by the machine or genotype precision. By means of this particular tournament selection procedure, the detrimental effect on 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 augmented against for the contingent average increase that may occur especially during the advanced stages of the convergence. The smaller total mean of the constraint violations implies improved convergence to the optimum.

For the other cases, namely

$X_2<X_1,\ X_1,X_2 < n_b$

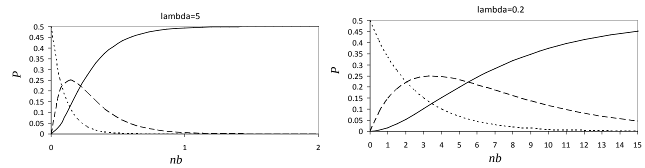
$$P(X_2 < X_1 < n_b) = 0.5(1-e^{-2\lambda_j n_b}) - e^{-\lambda_j n_b}(1-e^{-\lambda_j n_b}) \quad (41)$$

and for $X_2<X_1$, $n_b<X1,X2$

$$P(n_b < X_2 < X_1) = 0.5e^{-2\lambda n_b} \quad (42)$$

The variations of the different probabilities in (41)-(42) are plotted together in figure 11. It is to note that for any value of n_b, the summation of the probabilities is equal to 0.5, which conforms to (28) for $\lambda_1=\lambda_2$.

Figure 11 is the counterpart of figure 9 in the objective function space.



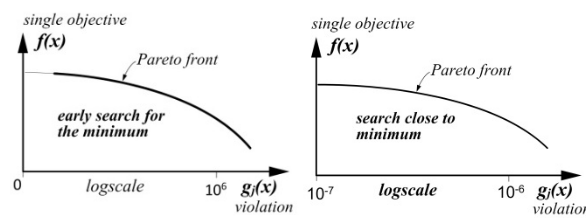
(a) (b)

Fig. 11. Plot of the probabilities for different conditions that can arise during binary tournament selection for λ=5 (a); λ=0.2 (b)

It is seen from figure 11 that the shape of the probability functions depends on λ_j, whereas in the probabilistic domain in figure 9, the shape remains constant, i.e. independent of λ_j.

C. Zooming for Robust Ranking

Zooming for robust ranking is accomplished by accurate ranking the favourable solutions between zero and unity as probabilistic distances, even though the actual constraint values may be close to the utopic optimal point as much as allowed by the computer precision that may be at the range of 10^{-10} or below. Illustration of the Pareto front at the early stage of the genetic search is given in figure 12a. Illustration of the Pareto front at the last stage of the genetic search is given in figure 12b.

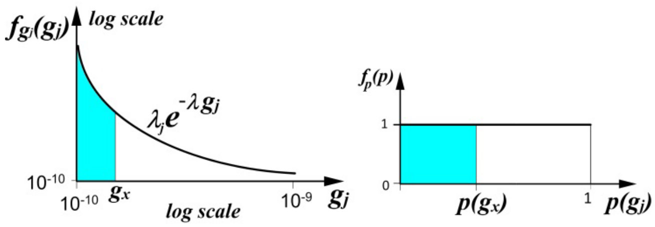


(a) (b)

Fig. 12. Illustration of the formation of the Pareto front at the early stage (a) and last stage of the genetic search (b)

Considering figure 12b, the probabilistic distance to the minimum is illustrated as a typical example in figure 13a by the shaded area where computation of the shaded area is very precarious at the tournament selection process. This is due to the issue of both exact parameterization of the exponential pdf in the existing range and the finite machine precision. This issue is prevented in figure 13b by taking simply $p(g_j)$ as the probability distance to the minimum. The marked areas in figure 13a and 13b are the same, and they are equal to $p(g_j)$.

The marked area in figure 13a, is represented in figure 13b 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 13b. This treatment is equivalent to applying a matching 'magnifying glass' to the space formed by actual objective function and the constraint functions, in order to carry out the convergence process without being affected by any scale of convergence present in this very space. The Pareto front at this micro scale is illustrated in figure 12.



(a) (b)

Fig. 13. An example illustration of the probability density of the constraint violations in the objective functions space (a) and the probabilistic distance space (b).

D. FAST AND ROBUST CONVERGENCE

With the probabilistic distance for nonlinear ranking we obtain an optimal step for convergence at each generation. To see this, from (27)

$$r_j = \frac{p(g_j)}{g_j} = \frac{1-\exp\left[-\lambda_j g_j\right]}{g_j} \approx \frac{1-\exp\left[-\frac{g_j}{\bar{g}_j}\right]}{g_j} \tag{43}$$

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

$$\lim_{g_j \to 0} r_j = \frac{p(g_j)}{g_j} = \lim_{g_j \to 0} \lambda_j e^{-\lambda_j g_j} = \lambda_j \to \infty \tag{44}$$

As it is seen, the genetic search algorithm is extraordinarily stable, that is, the convergence is due, and due to monotonic increase of the slope r_j, the convergence is fast.

IV. COMPUTER EXPERIMENT

Computer experiments have been carried out using a standard optimization problem from the literature. To demonstrate the robust, fast and accurate computations the course of the convergence are given in detail.

The following problem is due to Himmelblau [20]. given by

$$f(\boldsymbol{x}) = 5.3578547 x_3^2 + 0.8356891 x_1 x_5 + 37.293239 x_1 - 40792.141 \tag{45}$$

where the ranges for the independent variables are given by

$$78 < x_1 < 102;\ 33 < x_2 < 45;\ 27 < x_i < 45\ \ (i = 3,4,5) \tag{46}$$

subject to:

$$\begin{aligned}
&g_1(\boldsymbol{x}) = 85.334407 + 0.0056858 x_2 x_5 + 0.0006262 x_1 x_4 - \\
&0.0022053 x_3 x_5 - 92 \le 0 \\
&g_2(\boldsymbol{x}) = -85.334407 - 0.0056858 x_2 x_5 - 0.0006262 x_1 x_4 + \\
&0.0022053 x_3 x_5 \le 0 \\
&g_3(\boldsymbol{x}) = 80.51249 + 0.0071317 x_2 x_5 + 0.0029955 x_1 x_2 + \\
&0.0021813 x_3^2 - 110 \le 0 \\
&g_4(\boldsymbol{x}) = -80.51249 - 0.0071317 x_2 x_5 - 0.0029955 x_1 x_2 - \\
&0.0021813 x_3^2 + 90 \le 0 \\
&g_5(\boldsymbol{x}) = 9.300961 + 0.0047026 x_3 x_5 + 0.0012547 x_1 x_3 + \\
&0.0019085 x_3 x_4 - 25 \le 0 \\
&g_6(\boldsymbol{x}) = -9.300961 - 0.0047026 x_3 x_5 - 0.0012547 x_1 x_3 - \\
&0.0019085 x_3 x_4 + 20 \le 0
\end{aligned} \tag{47}$$

The problem consists of a single objective with 6 constraints, subject to minimization. The best known optimum is located at

$f(\boldsymbol{x}^*)=-30665.53867178332$

and the corresponding best variable values are

$x_1^*=78$; $x_2^*=33$; $x_3^*=29.9952560256815985$; $x_4^*=45$; $x_5^*=36.7758129057882073$.

The algorithm is executed with the following settings: population size=200; amount of generations=60; C=100000; crossover probability=0.9; mutation probability=0.05. The results are shown in figure 14-16 using a logarithmic scale for the horizontal axis, which shows the total violation G. It is noted that a single iteration of the algorithm consists of five generations.

After 5 iterations the population is seen in figure 12, where the best feasible solution is

$f(\boldsymbol{x})=-30569.5213239566$.

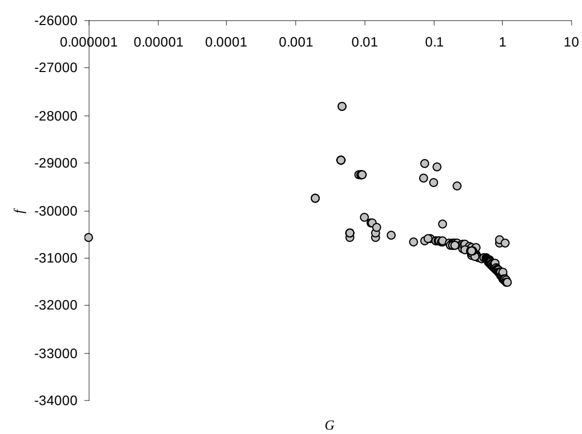


Fig. 14. Population after the 5-th iteration

The independent variables of this solution take:

$x_1=78.0265736760284$; $x_2=33.658770910086$; $x_3=30.470062623374$; $x_4=44.7895003744468$; $x_5=35.8616204529277$.

After 10 iterations the population is seen in figure 14, where the best feasible solution is found to be

$f(\mathbf{x})$=-30653.7876324169.

The independent variables of this solution take:

x_1=78.0261567922629; x_2=33.0152143977847; x_3=30.0228958188968; x_4=44.8232338738051; x_5=36.7924311001587.

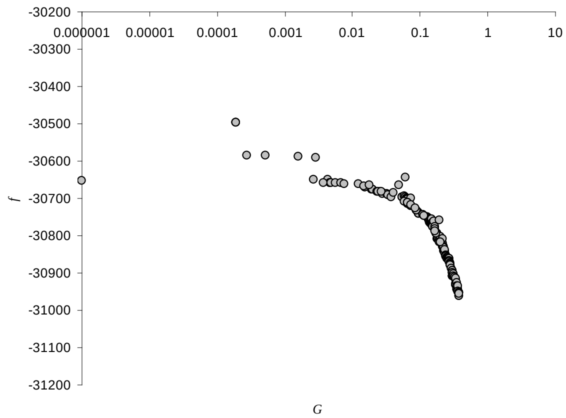


Fig. 15. Population after the 10-th iteration

After 30 iterations the population is seen in figure 15, where the best feasible solution is found to be

$f(\mathbf{x})$= -30665.4759429232.

The independent variables of this solution take:

x_1=78.0000867626641; x_2=33.000032984143; x_3=29.9955726882451; x_4=45; x_5=36.7751232308258.

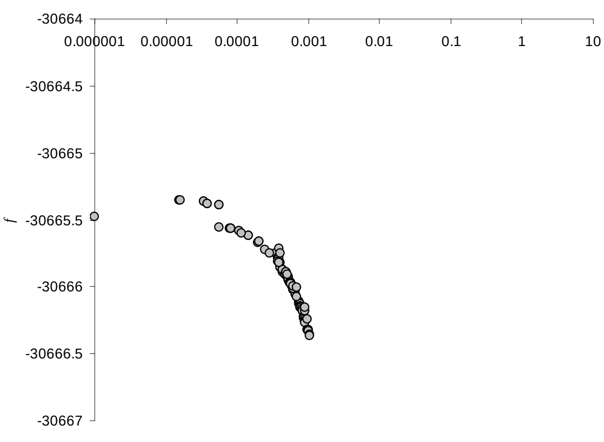


Fig. 16. Population after the 30-th iteration

It is noted that the process will continue to improve the solution more and more as the search continues, i.e. the population converges at the optimal solution, demonstrating the robustness of the approach. Namely after 60 iterations the population is seen in figure 16, where the best feasible solution is found to be

$f(\mathbf{x})$= -30665.5386683921.

The independent variables of this solution take

x_1=78.0000000039558; x_2=33.0000000083502; x_3=29.9952560418378; x_4=45; x_5=36.7758128740195.

V. Conclusions

Probabilistic considerations underlying a novel evolutionary computation are presented. In this work, multi-objective optimization is considered in the form of constraint optimization, the case conventionally being described in the literature, selecting appropriate penalty function parameters. However, since these parameters vary during the search process the determination of these parameters is very elusive and remained an issue to treat for researches. In contrast to this, in this work, a probabilistic model is introduced, by means of which the penalty parameters are embedded in the model, and they are inherently tuned, as the model is adaptively modified throughout the generations. The probabilistic model also has several favorable implications, which are treated in this research. These are stiffness handling, impartial elitism, zooming for robust ranking, as well as fast and robust convergence. The theory presented in this work is exemplified by an optimization problem for demonstration of the general effectiveness resulting from this analytical treatment of the constraint optimization methodology. However the method is not restricted to constraint optimization, but suitable for multi-objective optimization in general. The reported results include not only the final outcomes but also the progress of the convergence throughout the optimization process conforming exactly to the theoretical considerations presented.

References

[1] A. Homaifar, S. H. V. Lai, and X. Qi, "Constrained optimization via genetic algorithms," *Simulation,* vol. 9, p. 19, 1994.

[2] Z. Michalewicz and M. Schoenauer, "Evolutuionary algorithms for constrained parameter optimization problems," *Evolutionary Computation,* vol. 4, p. 33, 1996.

[3] Z. Michalewicz, *Genetic Algorithms+ Data Structures=Evolution Programs*. Berlin: Springer, 1999.

[4] S. Venkatraman and G. G. Yen, "A Generic Framework for Constrained Optimization Using Genetic Algorithms," *IEEE Trans. Evolutionary Computation,* vol. 9, p. 12, 2005.

[5] Z. Huang and H. Tian, "A genetic algorithm with constrained sorting method for constrained optimization problems," presented at the IEEE International Conference on Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. , Shanghai 2009.

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

[7] S. B. Hamida and M. Schoenauer, "An adaptive algorithm for constraint optimization problems," in *Lecture Notes on Computer Science PPSN 2000*. vol. 1917, K. Deb, Ed., ed Heidelberg: Springer, 2000, pp. 529-538.

[8] S. B. Hamida and M. Schoenauer, "ASHEA: New results using adaptive segregtional constraint handling," presented at the IEEE Congress on Evolutionary Computation (CEC 2002), Honolulu, Hawaii, 2002.

[9] F. Hoffmeister and J. Sprave, "Problem-independent handling of constraints by use of metric penalty functions," presented at the Fifth Annual conference on evolutionary programming EP 1996, San Diego, CA, 1996.

[10] H. Sarimveis and A. Nikolakopoulos, "A line up evolutionary algorithm for solving non-linear constraint optimization problems," *Computers & Operations Research* vol. 32, pp. 1499-1514, 2005.

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

[12] T. Ray, H. K. Singh, A. Isaacs, and W. Smith, "Infeasibility driven evolutionary algorithm for constrained optimization," in *Constraint-Handling in Evolutionary Optimization*, ed Berlin: Springer, 2009, pp. 145-165.

[13] K. Deb and R. Datta, "A fast and accurate solution of constrained optimization problems using a hybrid bi-objective and penalty function approach," presented at the Evolutionary Computation (CEC), 2010 IEEE Congress on, Barcelona 2010.

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

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

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

[17] 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.

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

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

[20] D. M. Himmelblau, *Applied Nonlinear Programming*: Mc-Graw-Hill, USA, 1972.

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.

Tehir Cetin Akinci received the BSc degree in Electrical Engineering. His master's and PhD degrees, Institute of Pure and Applied Science from Marmara University, Istanbul, Turkey. He is an associate professor in the Department of Electrical Engineering at Istanbul Technical University (ITU), Istanbul, Turkey. His research interests are signal processing, data mining, intelligent systems, ferroresonance phenomenon, artificial neural networks, and renewable energy sources.

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.

THE JOURNAL OF
COGNITIVE SYSTEMS
VOLUME 01 NUMBER 01 DECEMBER 2016

NOTE ON THE INFORMATION-THEORETIC ASPECT OF FUZZY NEURAL TREE

Özer Ciftcioglu[1], Michael S. Bittermann[2]

[1] Delft University of Technology, Delft, The Netherlands
[2] Maltepe University, Istanbul, Turkey

A note on the information theoretic aspect of the fuzzy neural tree (FNT) is presented. The detailed description of the FNT is given in an earlier work, where its information-theoretic aspect is heuristically mentioned but not elaborated because of some space limitation. The present note is to highlight this aspect of the tree as this is important in working of the tree with its knowledge-driven structure.

Index Terms — Fuzzy logic, neural tree, knowledge modeling, evolutionary computation, likelihood, probability possibility

I. INTRODUCTION

Fuzzy neural tree (FNT) structure is given in an earlier published work where its information-theoretic aspect is briefly mentioned [1]. The present note is to highlight this aspect of the tree as this is important in working of the tree with its knowledge-driven structure. The information-theoretic aspect of the FNT concerns the application of fuzzy concept to some concepts of Information Theory and using the result as knowledge in the tree structure. In this way, the FNT is driven by both assessments of soft issues as fuzzy memberships, and fuzzy membership of measurement data. The rest of the FNT structure is formed by the fuzzy information as knowledge source for the tree. The brief description of the FNT is intentionally presented here for the sake of the completeness of the note.

Neural tree concept and neuro-fuzzy computation is well established in the literature. In particular, neural tree concept is a kind of "free format" neural computation where layer–by-layer structure of neural network is relaxed as this will be shown shortly afterwards. In the realm of neuro-fuzzy paradigm, a neural network can be considered as fuzzy system in the sense of the non-linearity introduced at the neurons can be seen as fuzzy membership functions. Although such a view is appealing from the fuzzy system viewpoint, fuzzy interpretation of a neural network becomes formidably involved as the network is not a simple one. Therefore, a neural network is established generally by learning the input data without any recourse to fuzzy considerations. Then such structure is considered as non-parametric model. On the other hand a neural network can be established by some fuzzy considerations as a knowledge model and the same structure can be seen as a parametric model, depending on the input data in both cases. Even in this parametric model case some ranked structure of the neural network can be relaxed and the knowledge considered in this context can be the information provided by the inputs of the network. As it can duly be anticipated, in this fuzzy model the input data is represented in terms of information and this information is fuzzified being subject to fuzzy information processing.

The organization of the paper is as follows. Section II describes the structural and computational aspects of fuzzy neural tree, as well as its information-theoretic aspect. This is followed by conclusions.

Manuscript received September 30, 2016; accepted October 14, 2016.
Corresponding author: Özer Ciftcioglu (E-mail:: o.ciftcioglu@tudelft.nl).
Digital Object Identifier:

II. FUZZY NEURAL TREE

A. Structural and Computational Aspects

A neural tree can be considered as a feed-forward neural network that is organized not layer by layer but node by node. The nodes comprise nonlinear functions for processing the incoming information. In fuzzy neural networks, this nonlinear function is treated as a fuzzy logic element like membership function or possibility distribution. Therefore, fuzzy logic is integrated into a neural tree with the fuzzy information processing executed in the nodes of the tree. A generic description of a neural tree subject to analysis in this research is as follows. Neural tree networks are in the paradigm of neural networks with obvious similarities in their structures. A neural tree consists of terminal nodes that also referred to as leaf nodes, non-terminal nodes that are also referred to as internal or inner nodes, and weights associated with the connection links between the pairs of nodes. The non-terminal nodes are considered to be neural units, as the neuron type is an element introducing a non-linearity simulating a neuronal activity. In the present case, this element is a Gaussian function, which has several desirable features for the goals of the present study; namely, it is a radial basis function ensuring a solution, as well as the smoothness. At the same time it plays the role of possibility distribution in the tree structure, which is considered to be a fuzzy logic system as its

ISSN 2548-0650 © 2016 The Journal of Cognitive Systems (JCS). Personal use is permitted, but republication/redistribution requires JCS permission. See www.dergipark.gov.tr/jcs for more information.

outcome is based on fuzzy logic operations thereby providing associated reasoning. In a conventional neural network structure there is a hierarchical layer structure, where each node at the lower level is connected to all nodes of the upper layer nodes. However, this stipulation is very restrictive when a general system should be represented. Therefore, a more relaxed network model is necessary, and this is accomplished by a neural-tree, the properties of which are as defined above. An instance of a neural tree is shown in figure 1. Each terminal node, is labeled with an element from the terminal set $T=[x_1, x_2, \ldots, x_n]$ where x_i is the *i*-th component of the external input $\boldsymbol{x}$ which is a vector. Each link *(i,j)* represents a directed connection from node *i* to node *j*. A value w_{ij} is associated with each link. In a neural tree, the root node is an output unit and the terminal nodes are input units.

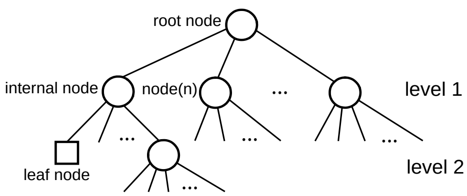


Fig. 1. Structure of a neural tree

A non-terminal node should have minimally multiple inputs to be meaningful, although a single input is also valid for operation. A node may have a single or multiple outputs;.

An internal node having a single input is considered to be a trivial case. This is because in this case output of the node is approximately equal to the input that it is to be considered equal. The node outputs are computed in the same way as computed in a feed-forward neural network. In this way, neural trees can represent a broad class of feed-forward networks that have irregular connectivity and non-strictly layered structures. In conventional neural tree structures generally connectivity between the branches is avoided. They are used for pattern recognition, progressive decision making, or complex system modeling. In contrast with such works, in the present research connectivity between the branches is possible, and the fuzzy neural tree structure is in a fuzzy logic framework for knowledge modeling, where fuzzy probability/possibility as element of soft computing is central. Added to this, the fuzzy neural tree functionality is based on likelihood representing fuzzy probability/possibility. This is another important difference between the existing neural trees in literature and the one in this work. Although in literature a family of likelihood functions is used to define a possibility as the upper envelope of this family [2, 3], to the authors' best knowledge there is no likelihood function approach in the context of neural tree. In the fuzzy neural tree, the output of *i*-th terminal node is denoted y_i and it is introduced to a non-terminal node. The detailed view of node connection from terminal node i to internal node j is shown in figure 2a and from an internal node i to another internal node j is shown in figure 2b.

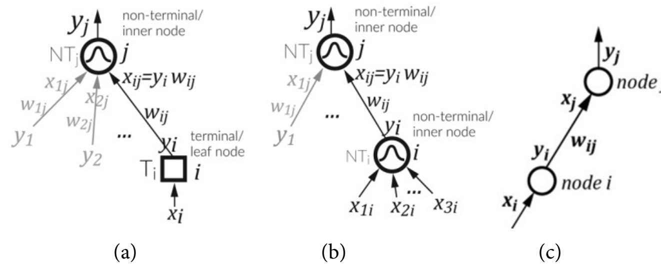


Fig. 2. The detailed structure of different type of node connections

The connection weight between the nodes is shown as w_{ij}. In the neural network terminology, a node is a neuron and w_{ij} is the synaptic strength between the neurons. This means, it represents the strength of connection between the nodes involved. In the fuzzy neural tree it is between zero and unity. Figure 3 shows some sample membership functions for the terminal nodes.

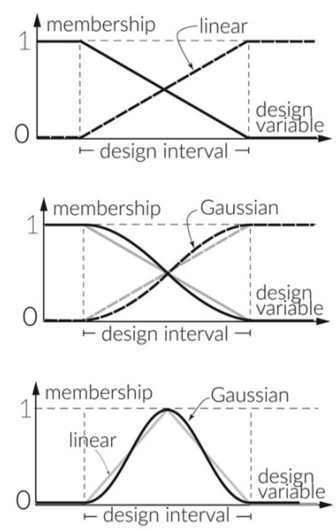


Fig. 3. Some sample membership functions at the terminal nodes

To start with we refer to figure 2a. We assume the input to an input node, namely a terminal node, is a Gaussian random variable, which is instructive to start with. In the fuzzy neural tree introduced in this work, all the processors operating in the internal nodes are Gaussian. Since the inputs to neural tree are also Gaussian random variables, due to functions of random variable theorem [4] all the processes in the tree are to be considered Gaussian. In a neural tree for each terminal input we define a linear or Gaussian fuzzy membership function as seen in figure 3, whose associated membership provides a probabilistic/possibilistic value for that input. Referring to figure 2, let us consider two consecutive nodes as shown in figure 2c. In the neural tree, any fuzzy probabilistic/possibilistic input delivers an output at any non-terminal node. Due to Gaussian considerations given above, we can consider this probabilistic/possibilistic input value of a node as a random variable x which can be modelled as a Gaussian probability density around a mean x_m. The probability density is given by

$$f_x(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(x - x_m)^2} \tag{1}$$

where x_m is the mean; σ is the width of the Gaussian.

Definition: Assuming a statistical model parameterized by a fixed and unknown θ the likelihood $L(\theta)$ is the probability of the observed data x considered as a function of θ.

The likelihood function of the mean value x_m is given by [5]

$$L(\theta) = e^{-\frac{1}{2\sigma^2}(x - \theta)^2} \tag{2}$$

where θ is the unknown mean value x_m. Likelihood function is considered to be as a fuzzy membership function or fuzzy probability, converting the probabilistic uncertainty to fuzzy logic terms. θ is a general independent variable of the likelihood function, and the likelihood is between *0* and *1*. $L(\theta)$ plays the role of fuzzy membership function and the likelihood at the node output is given by

$$y_j = L_j(\theta_j) \tag{3}$$

Referring to figure 2c, we consider the input x_j of node j as a random variable given by

$$x_j = y_i w_{ij} \tag{4}$$

where w_{ij} is the synaptic connection weight between the node i and node j seen in figure 2. In the same way as described above, the pdf of x_j is given by

$$f_{x_j}(x_j) = \frac{1}{\sqrt{2\pi}\sigma_j} e^{-\frac{1}{2\sigma_j^2}(x_j - x_{mj})^2} \tag{5}$$

and the likelihood function of the mean value $\theta = x_{mj}$ with respect to the input x_j is given by

$$L_j(\theta_j) = e^{-\frac{1}{2\sigma_j^2}(x_j - \theta_j)^2} = e^{-\frac{1}{2\sigma_j^2}(w_{ij} y_i - \theta_j)^2} \tag{6}$$

where θ is the likelihood parameter. Using (3) in (6), we obtain

$$L_j(\theta_j) = e^{-\frac{1}{2\sigma^2}(x_j - \theta_j)^2} = e^{-\frac{1}{2\sigma^2}(w_{ij} L_i(\theta_i) - \theta_j)^2} \tag{7}$$

We consider the neural tree node status where the likelihood is maximum for the input is maximum, namely $L_j(\theta_j)=1$ for $L_i(\theta_i)=1$. In (7) using $L_i(\theta_i)=1$ we obtain

$$\theta_j = w_{ij} \tag{8}$$

for the maximum likelihood $L_j(\theta_j)=1$. Hence, from (7) and (8), we obtain that likelihood $L_j(\theta_j)$ is maximum for $L_i(\theta_i)=1$ as was designed. $L_i(\theta_i)$ is the likelihood of the preceding node.

$$L_j(\theta_j) = e^{-\frac{1}{2\sigma_j^2}\theta_j^2 (L_i(\theta_i) - 1)^2} \tag{9}$$

Referring to (3), from (9) we can also write

$$L_j(\theta_j) = e^{-\frac{1}{2\sigma_j^2}\theta_j^2 (y_j - 1)^2} = e^{-\frac{1}{2\sigma_j^2}\theta_j^2 (1 - y_j)^2} \tag{10}$$

Referring to (9) the likelihood $L_j(\theta_j)$ is the probability of observed data as a function of θ via $L_i(\theta_i)$ which is the likelihood of the preceding node output. In other words, each likelihood output of a node is dependent on the probability of the outcome of the preceding node output, which is the observed data in this likelihood context.

B. INFORMATION-THEORETIC ASPECT

For $L_i(\theta_i) = 1$ the likelihood $L_j(\theta_j)$ is maximum being independent of θ_j. However for $L_i(\theta_i) \neq 1$, the likelihood $L_j(\theta_j)$ is dependent on θ_j. In (9), we note the variation of $L_j(\theta_j)$ with respect to θ_j while $L_i(\theta_i)$ is a parameter. For $L_i(\theta_i)$ close to unity or θ_j is close to zero likelihood, then $L_j(\theta_j)$ is close to maximum. From the information theory viewpoint, likelihood is probability p and the information is given by

$$I = -\log p = -\log L(\theta) \tag{11}$$

The information content of likelihood is given by (11) since $L(\theta)$ is considered to be a fuzzy probability [6] in the form of a membership function. The fuzzification of this information is accomplished by means of the information fuzzy membership function

$$MF = 1 - \exp(-I) \tag{12}$$

as this is shown in figure 4 with respect to information.

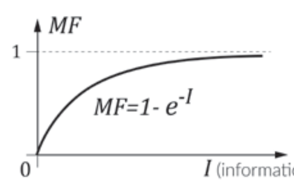


Fig. 4. Fuzzy membership function of information

The same information fuzzy membership function with respect to likelihood is shown in figure 5.

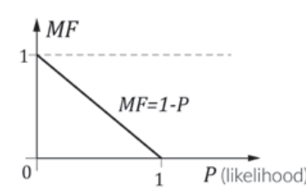


Fig. 5. Membership function of fuzzified information

The fuzzy membership function of information in figure 4 can take slightly different forms, taking the decay constant different than unity. In that case the membership function in figure 5 would read $MF = 1 - P^{\tau}$ where τ denotes the decay constant.

The membership function value of the fuzzified information is used as the connection weight in the fuzzy neural tree,

$$w_{ij} = 1 - p_{ij} = 1 - L_i(\theta_i) \qquad (13)$$

as was explained above by (1) through (8). The fuzzified information is to consider as *fuzzy information* between zero and unity. In the FNT, the connection weights throughout the model are determined by the inputs of the FNT without recourse to any expert knowledge, in this knowledge model. It is interesting to note that if the inputs of the model are measurement data, then the measurements are fuzzified by means of appropriate membership function to a fuzzy probability as shown in figure 6.

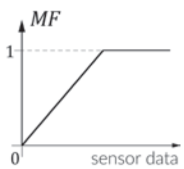


Fig. 6. Membership function as fuzzy probability

If the inputs of the model are soft inputs, then these inputs are considered to be directly fuzzified inputs between zero and unity and the fuzzfied information introduced above prevails throughout the model.

The heuristic explanation of (13) is as follows. θ_j refers to the connection of the node i to the node j. From the information theoretic viewpoint y_i is a probability and it contains no information when it is unity. In this case we do not have to convey any information from node i to node j, and therefore $\theta_j = 0$. From other side if y_i is zero, it contains information that it goes to infinity. Therefore, we connect the node i to node j with total connectivity, that means $\theta_j = 1$ in the case of single input. For a multiple input case, which is the non-trivial or actual situation, θ_j is selected in a normalized form for defuzzification in the rule-chaining process through from node to node process in the tree.

$$\theta_{ji} = \frac{1 - y_i}{\sum_{i=1}^{n} (1 - y_i)} \qquad (14)$$

$$y_j = L_j(\theta_j) = e^{-\frac{1}{2\sigma_j^2}\theta_j^2 (y_i - 1)^2} = e^{-\frac{1}{2\sigma_j^2}\left[\frac{1-y_i}{\sum_{i=1}^{n}(1-y_i)}\right]^2 (y_i - 1)^2} \qquad (15)$$

In (15) n is the index number of the number of inputs to the node j.

III. CONCLUSION

A note on a fuzzy neural tree is presented from the information-theoretic properties viewpoint involved in the tree. Information-theoretic viewpoint is essential for an automatic knowledge model formation directly from the inputs of the model. Heuristically we can consider that the information supplied to the model is from the inputs, and this information is used to form the model without any training process. This is an important property of the present neural tree structure since in the general neural tree concept in literature, the tree structure is determined in one way or other by learning and hence the model is non-parametric.

Fuzzy neural tree subject to study in this work, is an essential component of computational cognition, and its effectiveness is demonstrated in several applications reported in the literature [7, 8].

REFERENCES

[1] Ö. Ciftcioglu and M. S. Bittermann, "A Fuzzy Neural Tree Based on Likelihood," presented at the 2015 IEEE International Conference on Fuzzy Systems - FUZZ-IEEE 2015, Istanbul, Turkey, 2015.

[2] D. Dubois and H. Prade, "A semantics for possibility theory based on likelihoods," *J. of Mathematical Analysis and Applications* pp. 205, 359-380, 1997.

[3] D. Dubois and H. Prade, "A semantics for possibility theory based on likelihoods," in *Int. Joint Conf. of Fourth IEEE Int. Conf. on Fuzzy Systems and Second Int. Fuzzy Engineering Symposium - Fuzzy Systems 1995*, Yokohama, Japan, 1995, pp. 1597-1604.

[4] A. Papoulis, *Probability, Random Variables and Stochastic Processes*. New York: McGraw-Hill, 1965.

[5] Y. Pawitan, *In All Likelihood: Statistical Modelling and Inference Using Likelihood*. New York: Clarendon Press Oxford, 2001.

[6] L. A. Zadeh, "Is there a need for fuzzy logic?," *Information Sciences,* vol. 178, pp. 2751-2779, 2008.

[7] Ö. Ciftcioglu and M. S. Bittermann, "Computational Cognitive Color Perception," presented at the IEEE World Congress on Computational Intelligence - WCCI 2016, Vancouver, Canada, 2016.

[8] O. Ciftcioglu and M. S. Bittermann, "Fuzzy Neural Tree in Evolutionary Computation for Architectural Design Cognition," in *IEEE Congress on Evolutionary Computation - CEC 2015*, Sendai, Japan, 2015, pp. 2319 - 2326.

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.

Michael S. Bittermann was born in Germany in 1976. He graduated with the highest honors as MSc in Architecture from Delft University of Technology (TU Delft), The Netherlands, in 2003. He received his PhD degree in Design Informatics with the highest honors from the same university in 2009. Bittermann was distinguished as one of the five eminent researchers among all 2008-2009 PhD graduates of TU Delft in the form of the Young Researcher Fellowship of Delft University of Technology (80k€). He joined Maltepe University, Istanbul, Turkey in 2013, where he contributed to the research project titled 'Design by Cognition and Comprehension,' funded by the Scientific and Technological Research Council of Turkey. His research interest is computational cognition.

THE JOURNAL OF
COGNITIVE SYSTEMS

an international, peer-reviewed, indexed,
and open-access periodical

VOLUME 01, NUMBER 01

DECEMBER 2016

CONTENTS

www.dergipark.gov.tr/jcs

ISSN 2548-0650