

# PRECISION EVOLUTIONARY OPTIMIZATION PART II: IMPLEMENTATION AND APPLICATIONS

Michael S. Bittermann<sup>1</sup>, Tahir Cetin Akinci<sup>2</sup>, Ramazan Caglar<sup>2</sup>

<sup>1</sup> Maltepe University, Istanbul, Turkey

<sup>2</sup> 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 penalty 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: michaelbittermann@maltepe.edu.tr).  
Digital Object Identifier:

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  $0 \leq w_i$  for all objectives  $i=1, \dots, k$  so that a weighting problem can be stated as

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

### 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

$$\min f(\mathbf{x}) \quad \text{subject to } g(\mathbf{x}) = [g_1(\mathbf{x}), g_2(\mathbf{x}), \dots, g_m(\mathbf{x})]^T \leq 0 \quad (2)$$

We assume that the feasible region is of the form

$$S = \{x \in R^n \mid g(\mathbf{x}) = [g_1(\mathbf{x}), g_2(\mathbf{x}), \dots, g_m(\mathbf{x})]^T \leq 0\} \quad (3)$$

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

$$\min w_1 f(\mathbf{x}) + w_2 G(\mathbf{x}) \quad (4)$$

where

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

Thus, the problem definition becomes explicitly,

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

$$S = \{x \in R^n \mid g(\mathbf{x}) = [g_1(\mathbf{x}), g_2(\mathbf{x}), \dots, g_m(\mathbf{x})]^T \leq 0\}$$

With this formulation, the weighting method becomes appropriate to employ where  $w_1=1$ ,  $w_2=\mu$ . 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  $\varepsilon$ -Constraint method [13, 14].

### C. ISSUES OF THE PENALTY FUNCTION APPROACH

The problem statement given in (6) is written as

$$P(\mathbf{x}, R) = f(\mathbf{x}) + \sum_{i=1}^j R_j g_j(\mathbf{x}) \quad (7)$$

The function  $g_j(\mathbf{x})$  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

$$P(\mathbf{x}, R) = f(\mathbf{x}) + R \sum_{i=1}^j g_j(\mathbf{x}) \quad (8)$$

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.

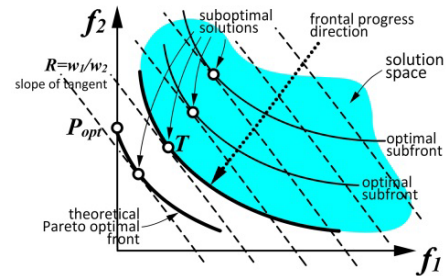


Fig. 1. Approach to the final optimal solution by means of penalty function approach;  $R$  is the penalty parameter

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  $f_2(\mathbf{x})$  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.

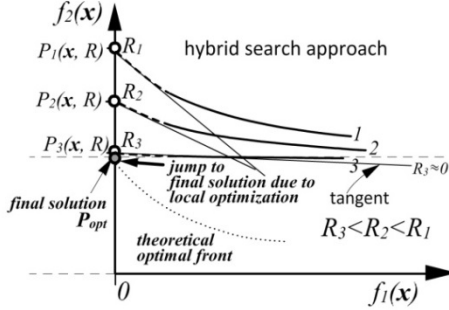


Fig. 2. Approach to the final optimal solution by means of penalty function approach;  $R$  is the penalty parameter

### 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

$$P(\mathbf{x}) = f(\mathbf{x}) + \sum_{i=1}^J \mu_i g_i(\mathbf{x}) \quad (9)$$

where  $f(\mathbf{x})$  is the single objective function to be minimized;  $g_j(\mathbf{x})$  is the violation of the  $j$ -th constraint, namely penalty function,  $\mu_i$  is the associated parameter of the penalty function. At each generation during the evolutionary minimization process  $g_j(\mathbf{x})$  is continually tried to be made to vanish. Considering the population density of solutions, this implies the probability density of  $g_j(\mathbf{x})$  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

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

where  $\lambda$  is the decay parameter. Denoting

$$y = g_j(x) \quad (11)$$

the pdf in (10) becomes

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

The mean value of the exponential pdf function is equal to  $\lambda_j^{-1}$ .

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

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

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

$$\lambda_j = 1/\bar{g}_j \quad (14)$$

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

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

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),  $\mu_j$  is replaced by  $Cr_j(g_j)$  in the form

$$Cr_j(g_j) = \mu_j(g_j) \quad (16)$$

So that (9) becomes

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

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

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

so that (17) becomes

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

In view of (18),  $r_j$  is given by

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

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.

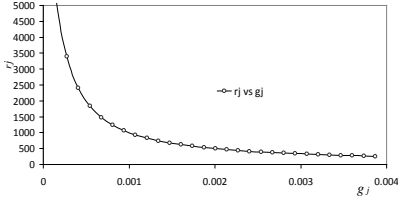


Fig. 3. Illustration of the new penalty parameter  $r$  as to probabilistic modeling:  $r=(1-\exp(-\lambda g))/g$

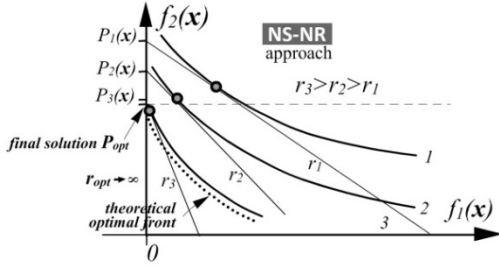


Fig. 4. Approach to the final optimal solution by means of penalty function approach;  $r$  is the penalty parameter.

#### IV. IMPLEMENTATION OF THE EVOLUTIONARY ALGORITHM

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

$$P(g_j, \mathbf{x}) = f(\mathbf{x}) + C \sum_{j=1}^J p(g_j(\mathbf{x})) \quad (21)$$

where  $J$  is the number of constraints;  $C$  is a common constant. The probability  $p(g_j)$  controls the penalty parameter,  $\mu_i(g_i)$  which is absorbed in  $p(g_j)$  in the form of  $r_j$ . The penalty parameter  $\mu_i(g_i)$  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.  $0 \leq p(g_j) \leq 1$  implies a tendency for aggregation in the space formed by  $f(\mathbf{x})$  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,

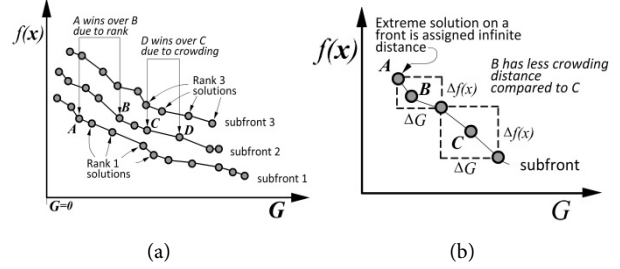


Fig. 5. Non-dominated sorting based selection among the infeasible solutions (a); Crowding distance computation (b)

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 subfronts remain large during the ranking-based front formation. Solutions in a tournament will be evaluated depending on the condition given by

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

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

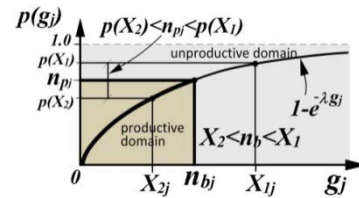


Fig. 6. Sketch for the selection procedure during non-dominated sorting (NS) based tournament

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 the 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  $p(g_i)$ -dimension, whereas in  $g_i$ -dimension the location of the boundary generally changes from generation to generation due to changing mean values.

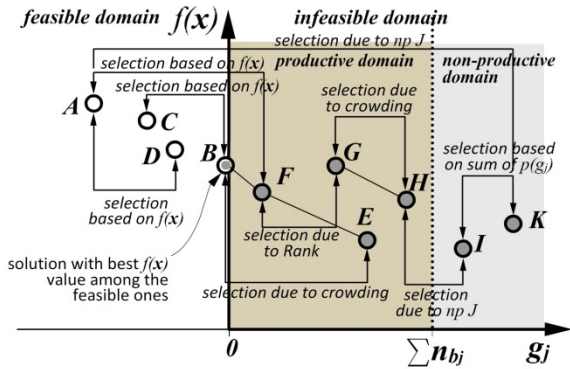


Fig. 7. Sketch for the tournament selection during NS

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  $f(x)$  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  $f(x)$ , 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  $f(x)$ , 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  $f(x)$  values. The feasible

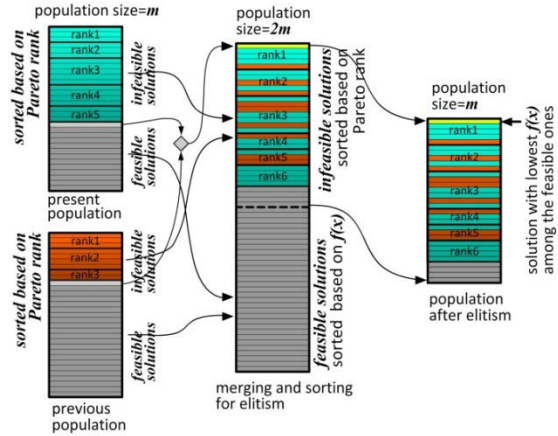


Fig. 8. NS based elitism

solutions with the lowest  $f(x)$  values are used to fill up remaining places in the elitist population. The feasible solution with the lowermost value of  $f(x)$  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  $P(g_j, \mathbf{x})$  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.

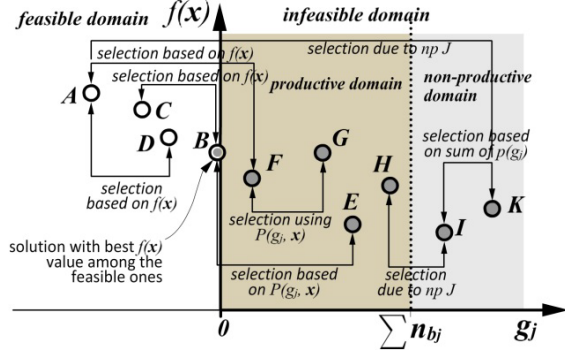


Fig. 9. Sketch of the tournament selection during NR

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  $P(g_j, \mathbf{x})$ . 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  $P(g_j, \mathbf{x})$  is a summation of function value  $f(\mathbf{x})$  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 [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,  $P(g_j, \mathbf{x})$  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  $P(g_j, \mathbf{x})$  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  $P(g_j, \mathbf{x})$  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.

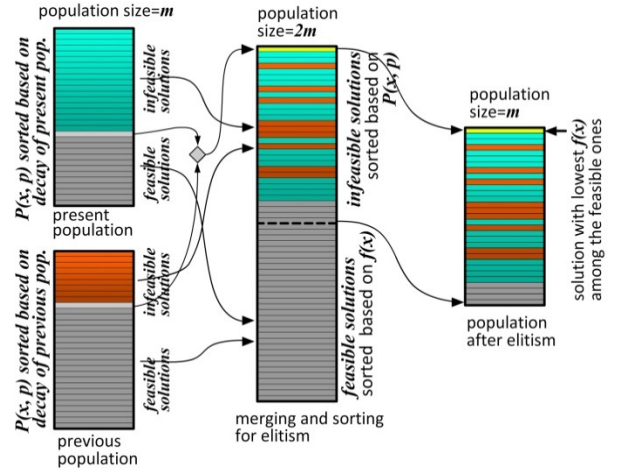


Fig. 10. NR based Elitism

## 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).

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

where the ranges for the independent variables are given by

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

Subject to:

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

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

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

The corresponding best known variable values are  
 $x_1^*=2.33049935147405174$ ;  $x_2^*=1.95137236847114592$ ;  
 $x_3^*=-0.477541399510615805$ ;  $x_4^*=4.36572624923625874$ ;  
 $x_5^*=-0.624486959100388983$ ;  $x_6=5$ ;  $x_7^*=1.03813099410962173$ ;  
 $x_8^*=1.5942266780671519$ .

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.

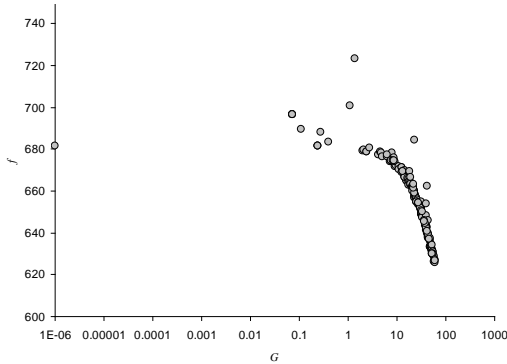


Fig. 11. Population after the 10-th iteration

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:

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

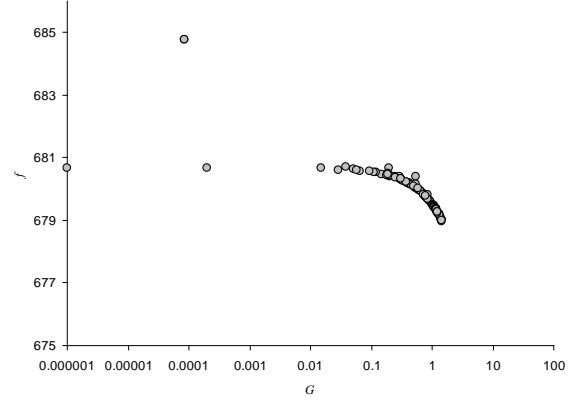


Fig. 12. Population after the 30-th iteration

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:

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

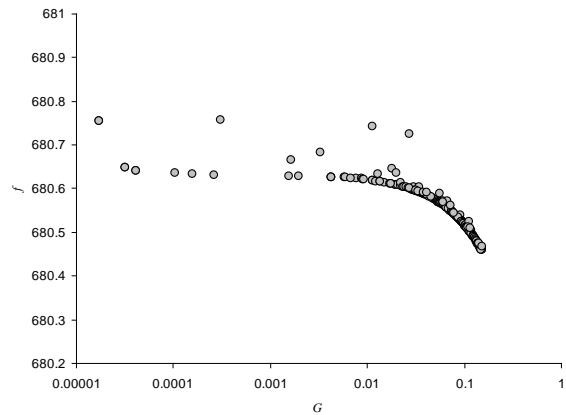


Fig. 13. Population after the 70-th iteration

## B. PROBLEM II

The following problem is due to Floudas 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).

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

where the ranges for the independent variables are given by

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

subject to:

$$g_1(\mathbf{x}) = -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 \leq 0 \quad (28)$$

$$g_2(\mathbf{x}) = (x_1 - 6)^2 + (x_2 - 5)^2 - 82.81 \leq 0$$

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

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

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

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

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

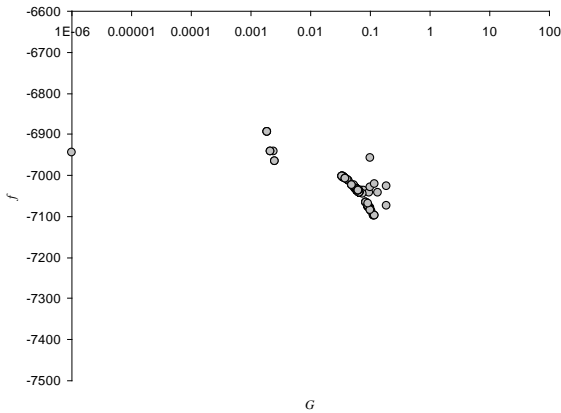


Fig. 14. Population after the 30-th iteration

After 100 iterations the best feasible solution is found to be

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

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

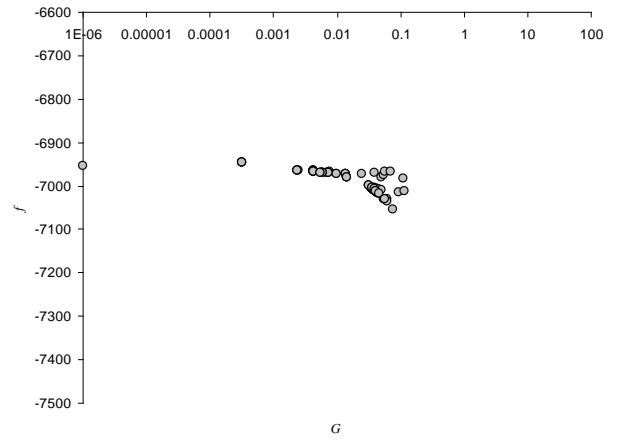


Fig. 15. Population after the 50-th iteration

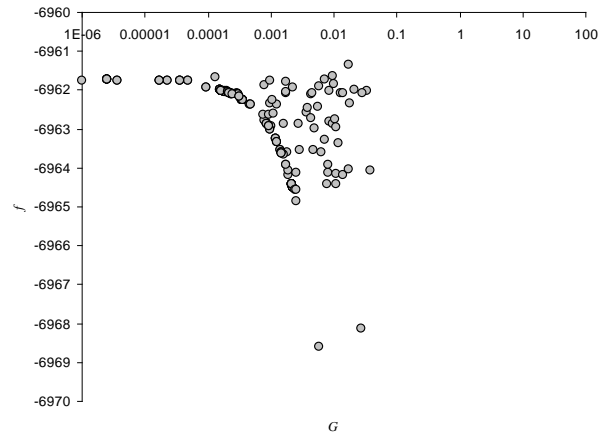


Fig. 16. Population after the 100-th iteration

## 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," *Ci Computers 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. Floudas 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).