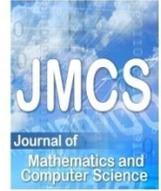


Contents list available at JMCS

Journal of Mathematics and Computer Science

Journal Homepage: www.tjmcs.com



A New Multi-Objective Sorting Algorithm and Its Combination with Game Theory for Optimizing I-beam Engineering System

Hamidreza Navidi¹, Mojtaba Ahmadi Khanesar², Leila Falahiazar³

¹ Associate Professor, Department of Applied Mathematics, Shahed University, Tehran, Iran.

² Ph.D. Candidate in Artificial intelligence, Islamic Azad University, Science and Research Branch, Computer Engineering.

³ Assistant Professor, Electrical and Control Engineering Department, Semnan University Semnan, Iran.

¹navidi@shahed.ac.ir, ²L.falahiazar@srbiau.ac.ir, ³ahmadi@profs.semnan.ac.ir

Article history:

Received September 2014

Accepted January 2015

Available online January 2015

Abstract

This paper proposed a new average non-dominated sorting genetic algorithm (NAVNSGA). This idea is inspired from the combination of non-elitist multi-objective evolutionary algorithms, elitist multi-objective evolutionary algorithms, and statistical calculations. The proposed NAVNSGA is improved the disadvantages of the Elitist multi-objective algorithms and Non-elitist multi-objective algorithms as possible. The NAVNSGA is compared with useful algorithms such as the non-elitist sorting genetic algorithm (NSGAI) and non-elitist sorting genetic algorithm (NSGAI) and the results obtained are showed the superiority of the proposed algorithm. Additionally, the NAVNSGA algorithm is combined with the concepts of the Game theory to propose a hybrid algorithm for determining Nash equilibrium in the game theory. The combination of the NAVNSGA algorithm with the game theory previously is used for improving engineering systems, such as I-beam designing. The results obtained are showed the advantage of the proposed algorithm with those reported in the literature.

Keywords: Multi-objective evolutionary algorithms, NSGAI, NSGAI, Game theory.

1. Introduction

Nowadays, artificial intelligence is used for solving real-world problems [1], [2], [3], [4]. Classical methods do not solve Multi-objective problems which makes proposing the new methods necessary. There are various methods for solving multi-objective problems. In the past decade, the evolutionary

algorithms have been very useful [5]. Evolutionary algorithms are used in designing and implementing many engineering problems. Recently, multi-objective evolutionary algorithms are used to solve problems in game theory. Some applications have been developed for simulating the combination of evolutionary algorithms and the game theory [6]. One of the important problems in game theory is determining a Pareto front. If the expectation for selecting one solution of the Nash equilibrium is higher than other, this point is called a “focal point” [7]. Finding “focal point” in game theory for solving problems in the real world is a challenge subject. Therefore, this brings us to combine the NAVNSGA Algorithm with the game theory. On the other hand, the NAVNSGA algorithm is designed to concentrate some solutions that are balanced more among other Nash equilibrium points. Multi-objective evolutionary sorting algorithms have been used in finding solutions of many multi-objective problems in the real world. They have been also widely used in optimizing engineering systems [8], [9], [10], [6], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21]. On the other hand, the development of multi-objective optimization algorithms used in many scientific fields is unequivocal. On the other hand, solving the disadvantage of multi-objective sorting algorithms (the NSGAI is the most well-known algorithm) leads us into to optimize many scientific problems [22], [23], [24], [25]. The algorithm proposed in this paper, the NAVNSGA algorithm tries to improve the disadvantage of NSGAI and NSGAI algorithms combinatorial to achieve balanced solutions. Then, the NAVNSGA algorithm illustrated for designing an I-Beam, the results obtained are compared with NSGAI and NSGAI algorithms and it is better than those reported in the literature.

2.1. The general form of multi-objective functions

The general form of multi-objective function is defined as follows [26], [27] :

Find a vector X with $\vec{X} = \{X_1, X_2, \dots, X_N\}$ components; so, the functions $f_i(\vec{X})$, $i = 1, 2, \dots, k$ can be minimized and constrained to the following conditions:

$$f_i(\vec{X}), i = 1, 2, \dots, k \quad (1)$$

Subject to:

$$g_j(X) \leq 0$$

$$X_i^{\min} \leq X_i \leq X_i^{\max}$$

The combination of evolutionary algorithms for multi-objective optimization problems using game theory approach is much simpler than classical methods.

2.2. Non-elitist multi-objective evolutionary algorithms

The most well-known and widely used algorithm in this field is NSGAI [26]. The algorithm can be outlined as follows (Algorithm 1):

Algorithm 1. The standard NSGAI

1. [Start] initial a random population (P0).
2. [Fitness] Evaluate the multi-objective fitness of each chromosome x in the population.
3. [Rank] Rank population by following steps:

- 3-1. [Domination Rank] Rank population by concept of domination [26].
- 3-2. [Fitness Sharing] Calculate the Fitness Sharing by using Algorithm 2.

In this algorithm, the fitness of each individual is the value of Fitness Sharing function. The Fitness Sharing function can be described as follows (Algorithm 2):

Algorithm 2. The Fitness Sharing function

- 1) Set $F_{min}=N+\epsilon$.
- 2) For every individual in this subset do these five steps:
 - 2-1. Set fitness for all individuals $F_i=F_{min}+\epsilon$.
 - 2-2. $d_{ij}=\sqrt{\sum_{k=1}^n (f_k^i - f_k^j)^2}$
 - 2-3. $SH(d_{ij}) = \left\{ 1 - \left(\frac{d}{\sigma_{share}} \right)^\alpha \right\}$
 - 2-4. $nc_i = \sum_{j=1}^N SH(d_{ij})$
 - 2-5. $f_i = \frac{F_i}{nc_i}$
- 3) Delete this sub set of population.
- 4) Set F_{min} with minimum fitness in this sub set.

The main disadvantage of the NSGAI is gradually tending to specific points in Pareto front and such a feature is used in the NAVNSGA and the NAVNSGA algorithm use it for finding a Nash equilibrium of the Game Theory [28].

2.3. Elitist multi-objective evolutionary algorithms

One of the most widely used algorithms in this field is the NSGAII (Algorithm 3) [25]. Its procedures are given as follows:

Algorithm 3. The standard NSGAII.

1. Generate a random population of N chromosomes.
2. Evaluate the multi-objective fitness of each chromosome.
3. Rank population by following steps:
 - 3.1. Rank population by using Dominance concepts [26].
 - 3.2. Calculate the crowding distance by using Algorithm 4.
4. Create a new population by repeating the following steps until the new population is complete.
 - 4-1. Select two parent chromosomes from a population.
 - 4-2. Crossover two parents.
 - 4-3. Mute new offspring.
 - 4-4. Place new offspring in the new population.
5. Replace new generated population.
6. If the end condition is satisfied; else, Go to step 2.

The crowding distance Function (algorithm 4) is given as follows:

Algorithm 4. Crowding distance.

1. Let $d_i = 0$ for $i = 1, 2; \dots, Z$.
2. Sort the set Ascending.
3. Let d_1 and d_Z be maximum values distance, e.g. $d_1 = \infty, d_Z = \infty$.
4. For $j = 2$ to $(Z - 1)$, set $d_j = d_j + (f_{kj+1} - f_{kj-1})$.

The first disadvantage encountered in the NSGAI algorithm is that individuals may be repeated numerously in a new population, since they belong to a higher-rank set. In fact, NSGAI is faced with the problem of slow convergence [27], [25]. The second problem is that, if the size of the first non-dominated set is bigger than the size of population then the convergence decreases and individuals tend to an extremum value of objective functions [26].

3.1. Proposing a New Average Non-dominated Sorting Genetic Algorithm

A New Average Non-dominated Sorting Genetic Algorithm (NAVNSGA) is proposed in this paper to solve the disadvantage of NSGAI and NSGAI algorithms that discussed later. Therefore, they were explained in detail in previous sections. The NAVNSGA algorithm is similar to many multi-objective sorting algorithms, but differs in some details. There are three sub-algorithms or innovations in the NAVNSGA algorithm which are discussed later. Each innovation is a heuristic approach for increasing the performance of searching procedure of the NAVNSGA algorithm.

3.2. The first innovation in the NAVNSGA algorithm

The first innovation in the NAVNSGA algorithm is that mute and cross operators to explore and to exploit by using of the non-dominated sub-set. Thus for decreasing diversity, we need a new approach which is explained below.

New cross operator works as follows:

- A chromosome in the dominated set can cross with a chromosome from the non-dominated set with a probability of cross rate and generate a new child.

New mute operator works as follows:

- A chromosome in the dominated sub-set can mute with a chromosome from the non-dominated set with a probability of mute rate and generate a new child.
- ✓ The non-dominated set is crossed and muted randomly.

3.3. The second innovation in the NAVNSGA algorithm

As mentioned earlier, the NSGAI algorithm obtains an early convergence which tends toward the non-dominated set. The following statistical procedure is explained to solve this disadvantage (the i is index of non-dominated set, the j is index of dominated set).

$$\left(\sum F_{\text{non-Dominated}-i} / \text{the size of the non-Dominated set}\right) - \left(\sum F_{\text{Dominated}-j} / \text{the size of the dominated set}\right) \ll \text{Low} \quad (2)$$

The Low value is determined by and related to the problem. The equation (2) is a statistical method that uses for increasing diversity in [29]. If the condition in Eq. (2) is satisfied, some genes of chromosomes in the non-dominated set (the first class) will be randomly reset. In fact, it is a Hill-Climbing method that is using for blind search [29]. The procedure of the NAVNSGA algorithm is as follows (Algorithm 5):

Algorithm 5. The NAVNSGA algorithm.

- 1) The basic population (P_{t0}) is randomly initialized.
- 2) Children form through parent population using New Mute and New Cross.
- 3) P_t is then concatenate with children population (C_t).
- 4) P_t+C_t is divided into non-dominated sets (Like NSGAI).
- 5) Sub-sets are then added to the new population by the order of non-dominated concept. The first sub-set, sub-set 1, which dominates other sub-sets is added to the population, and then the second set is added, etc.
- 6) Is the size of the last sub-set for adding the new population more than whole remaining size? If “Yes”; go on to the next step. If “No”; go to step (9).
- 7) First, Members of the last set is sorted with New Fitness Sharing algorithm.
- 8) The sorted sub-set (by the order of fitness) is added to the end of the population.
- 9) Average f_i (for each f_i) is calculated.
- 10) Is $\left(\sum F_{\text{non-Dominated}-i} / \text{the size of the non-Dominated set}\right) - \left(\sum F_{\text{Dominated}-j} / \text{the size of the dominated set}\right) \ll \text{Low}$ satisfied?
 If “Yes,” Then go to step (11).
 If “No,” Then go to step (12).
- 11) The sub-set of the non-dominated with a strong Mute Rate is mutated.
- 12) Are End conditions satisfied?
 If “Yes”; Then go to step (13).
 If “No”; Then go to step (2).
- 13) End.

In step (10) when distance of best solution in each objective function is very near to average of population, it means the diversity decreased, so we use a strong mutation to increase diversity. The New Fitness Sharing algorithm explained later.

3.4. The third innovation in the NAVNSGA algorithm

As considered in the previous algorithm (Algorithm 5), the P_t+C_t population is first classified based on the dominance concept. However, if the size of remaining individuals in the last set that must add within new population is more than the size of whole remaining individuals of the new population, the last sub-set will be sorted based on the New Fitness Sharing

algorithm(This method is similar to niche count method of NSGAI). Then, the sorted sub-set adds to the end of the new population. The New Fitness Sharing algorithm have the same operators with the Niche Count, but the reason of proposed the New Fitness Sharing algorithm is the advantage of the NSGII; when the number of generations increases, all individuals lie on Pareto front and they are distributed on the Pareto Front(investigated by Deb [26]). In many real-word problems; however, the aim is to achieve a Nash equilibrium (“focal point” [7]), or a limited space in Pareto front. When the number of generation increases, using the Niche Count method is concentrated solutions. In this way, more feasible number of suggestions are calculated for balancing objective functions with respect to one another. In fact, when the number of generation increases, using New Fitness Sharing Algorithm caused solutions do not tend towards maximum or minimum values of the objective function; rather, they tend to objective functions with more balanced value, and we have the special area in Pareto Front.

3.5. A New Fitness Sharing Function

The New Fitness Sharing function is different from the sharing function proposed by Deb in [26]. The algorithm is comprised of the following steps:

Algorithm 6. The New Fitness Sharing function algorithm.

1) The rank number of a subset (last class) is assigned to fitness value of individuals in last sub non-dominated ($f_{Initial\ fitness}$).

2) Obtain a niche count from the following equation ($i, j = 1, 2, \dots$, The number of sub set members)

$$2-1. d_{ij} = \sqrt{\sum_{k=1}^n (f_k^i - f_k^j)^2}$$

$$2-2. SH(d_i) = \left\{ 1 - \left(\frac{d}{\sigma_{share}} \right)^\alpha \right\}$$

$$2-3. nc_i = \sum_{j=1}^N SH(d_{ij})$$

3) Obtain individual's fitness from the following equation:

$$3-1. F_i = \frac{f_{Initial\ Fitness}}{nc_i}$$

3) Sort the sub-set based on fitness.

4) Fill the end of the new population sequentially by this sub-set.

According to the procedures of the New Fitness Sharing function, the fitness of individuals in the last class is determined in step 3. Afterwards, individuals are sorted based on their assigned fitness and then the sorted array is added to the remaining of new population by order. The New Fitness Sharing function is also different from individual selection method for the new generation in NSGAI.

4.1. Game theory approach

In the game theory, there are two types of games: the non-cooperative games and the cooperative games [5], [14], [15], [30], [31]. A basic understanding of those games is that, the number of design variables (N) are equal of number of objective functions (K). In the game theory, each

objective function corresponds as a player. Therefore, in problems which the goal is to find the minimum, the i th player tries to minimize its corresponding function f_i . Moreover, for the sake of simplicity, the i th player just controls x_i variable in the problem, only one variable (It should not affect any other variable). So, the i th player tends to minimize its corresponding function $f_i(x_1, x_2, \dots, x_i, \dots, x_n)$. Other players can affect the value of f_i through controlling other variables in the problems, such as $x_1, x_2, \dots, x_{i-1}, x_{i+1}, \dots, x_n$. In other words, the i th player can control the value of objective functions corresponding to other players, which are $f_1, f_2, \dots, f_{i-1}, f_{i+1}, \dots, f_n$, through controlling its own corresponding variable which is x_i . The player who “announces” its move first, in order to selecting an appropriate value for its variable can force other players to play in its own selected equilibrium point. A feasible solution \vec{X} for a multi-objective problem is called Pareto optimal point that is defined as follows, there is no other feasible solution like \vec{Y} such that $f_i(\vec{Y}) \leq f_i(\vec{X})$ for $i=1,2,\dots,k$ with $f_j(\vec{Y}) < f_j(\vec{X})$ for at least one j [10]. The first task in cooperative games is finding Pareto optimal set or Pareto space (SP). All points on one solution space can be considered as Pareto front points. Therefore, each point in this area is characterized by not being dominated by any other points in its surrounding. Then, the second task is to find a specific element (a parameters vector) from SP which represents a compromise solution which is acceptable for all players. This point is the “focal point” mentioned earlier. In order to achieve this, players determine specific rules for negotiations which can be used for formulating a super-criterion condition or a bargaining model. Then, the super-criterion may be used for converting a multi-objective optimization problem into only a single-objective optimization problem, this solution which makes a compromise solution (focal point) accepted by all players (an element from Sp or Pareto optimal set). Finding a Pareto optimal set Sp and also finding a unique solution (an element from Sp) based on multi-objective constraints have been widely investigated under the study of Game theory. Hybrid and improved methods of multi-objective evolutionary algorithms and Game theory are attracted attention for optimization problems, because these methods are simple computation.

4.2. The combination of Game Theory and New AVerage Non-dominated Sorting Genetic Algorithm

In the proposed hybrid (cooperative) Game theory, all players compromise on one solution using a mutually compromised bargaining model or super-criterion model. In this study, the multi criterion is defined in such a way that deviation of them for the i -th objective function gives the worst value of the i -th objective function (or maximum possible value) for each player ($i = 1, 2, \dots, k$). A modified game theory approach by [15] presented, the worst value of objective functions for each particle (i -th) that is belong to f_i , is computed with a maximum / minimum value of objective functions. However, other objective vectors have their own optimal values (Particles that are under constraints are optimized with no regard to other objective functions), Such a procedure may run in a long presence of opposite objective function in nature, When an objective function increases, other functions decrease. Generally, in engineering applications, it could not be assumed that all objective functions are opposite of one another. Therefore, for optimizing multi-objective functions, alternative methods are needed for selecting the least desirable value for each objective function in order to carry out a modified game theory. A modified version of game theory was proposed here in which selecting the least (min) and most (max) desirable value for each objective function is carried out differently. In [5], a revised version of game theory is proposed in which finding the maximum or minimum values of physical (objective) functions is carried out using particle swarm optimization (PSO). Then, the

maximum and minimum values obtained from physical functions are given to game theory in order to find Pareto optimal solution. In the hybrid method proposed here; however, the problem of finding maximum and minimum values for physical functions does not exist. The algorithm presented later is a combination of the new version of the game theory with the NAVNSGA algorithm. The hybrid version of the Game theory using the NAVNSGA algorithm can be described as follows:

- 1) Minimize each k objective by Eq.(1) and subject to the constraint of Eq.(1) using the NAVNSGA algorithm. Find optimal points corresponding to objective functions.
- 2) Pareto optimal solutions, that are obtained from the NAVNSGA algorithm, could be combined with one another in the form of a weighted set of objective functions (FC) as follows:

$$FC = C_1 f_{n1}(\vec{X}) + C_2 f_{n2}(\vec{X}) + \dots + C_{k-1} f_{nk-1}(\vec{X}) + (1 - C_1 - C_2 - \dots - C_{k-1}) f_{nk}(\vec{X}) \quad (3)$$

Then, minimize FC based on the constrain stated by all combinations of $C_1, C_2, C_3, \dots, C_{k-1}$

and $C_k = 1 - c_1 - c_2 - \dots - c_{k-1}$ providing that $0 \leq c_i \leq 1$ and $\sum_{i=1}^k c_i = 1$.

Super-criterion of S equation ensures that if any function is normalized, gets as far as possible from the least desirable point (normalized), which is equal to 1 for $i = 1, 2, \dots, k$. Therefore, S is defined as:

$$S = \prod_{i=1}^k [1 - f_{ni}(\vec{X})] \quad (4)$$

So the formulation of a new objective function $F(Y)$ is generated, in order to find an optimal Pareto point which shows a compromise solution. Thus we have the following equation:

$$F(\vec{Y}) = FC - S \quad (5)$$

so that:

$$\vec{Y} = \{x_1, x_2, \dots, x_n, C_1, C_2, \dots, C_{k-1}\}^T, \text{ whit } 0 \leq c_i \leq 1, i = 1, 2, \dots, k - 1 \quad (6)$$

- 3) For finding Y , which is the best multi-objective compromise solution, $F(Y)$ should be minimized. $F(Y)$ minimizes Eq.(1) and its constrain; so that, it is appropriate for a mutually compromising bargaining model. Therefore, solutions are obtained as a set by using the NAVNSGA algorithm.

5. Design of an engineering system

An I-beam problem is an engineering design problem with multi-objective functions. It was solved using the combination of the NAVNSGA algorithm and game theory to evaluate the proposed approach. The optimal design of I-beam with two objective functions using the combination of the Game theory and the NAVNSGA algorithm is illustrated below.

5.1. Design of an I-beam

An I-beam design is shown in Fig. (1) [5]. Its parameters should be determined, so that it can tolerate pressures P and Q as seen in Fig. (1). Optimization problem was formulated as follows:

Find vector $\vec{X} = \{X_1, X_2, X_3, X_4\}$ to minimize (f_1, f_2) (7)

so that Eq. (8) minimizes its cross-section area (the amount of used material):

$$f_1(\vec{X}) = 2x_2x_4 + x_3(x_1 - 2x_4) \tag{8}$$

Eq. (9) shows failure rate (bending rate) in the middle of the bar resulting from vertical pressure P and horizontal pressure Q (pressure tolerance at rotation formula), which is given in Eq. (9):

$$f_2(\vec{X}) = \frac{PL^3}{4E(x_3(x_1 - 2x_4))^3 + 2x_2x_4(4x_4^2 + 3x_1(x_1 - 2x_4))} \tag{9}$$

subject to stress constraint:

$$\frac{M_Y}{Z_Y} + \frac{M_Z}{Z_Z} \leq \sigma_b \tag{10}$$

where Eq. (10) M_Y and M_Z are maximum bending moment along Y and Z, respectively. σ_b is the acceptable tolerance for the used material. Stress constraint could be written as follows:

$$\frac{180000x_1}{x_3(x_1 - 2x_4)^3 + 2x_2x_4[4x_4^2 + 3x_1(x_1 - 2x_4)]} + \frac{15000x_2}{(x_1 + 2x_4)x_3^3 + 2x_4x_2^3} \leq 16 \tag{11}$$

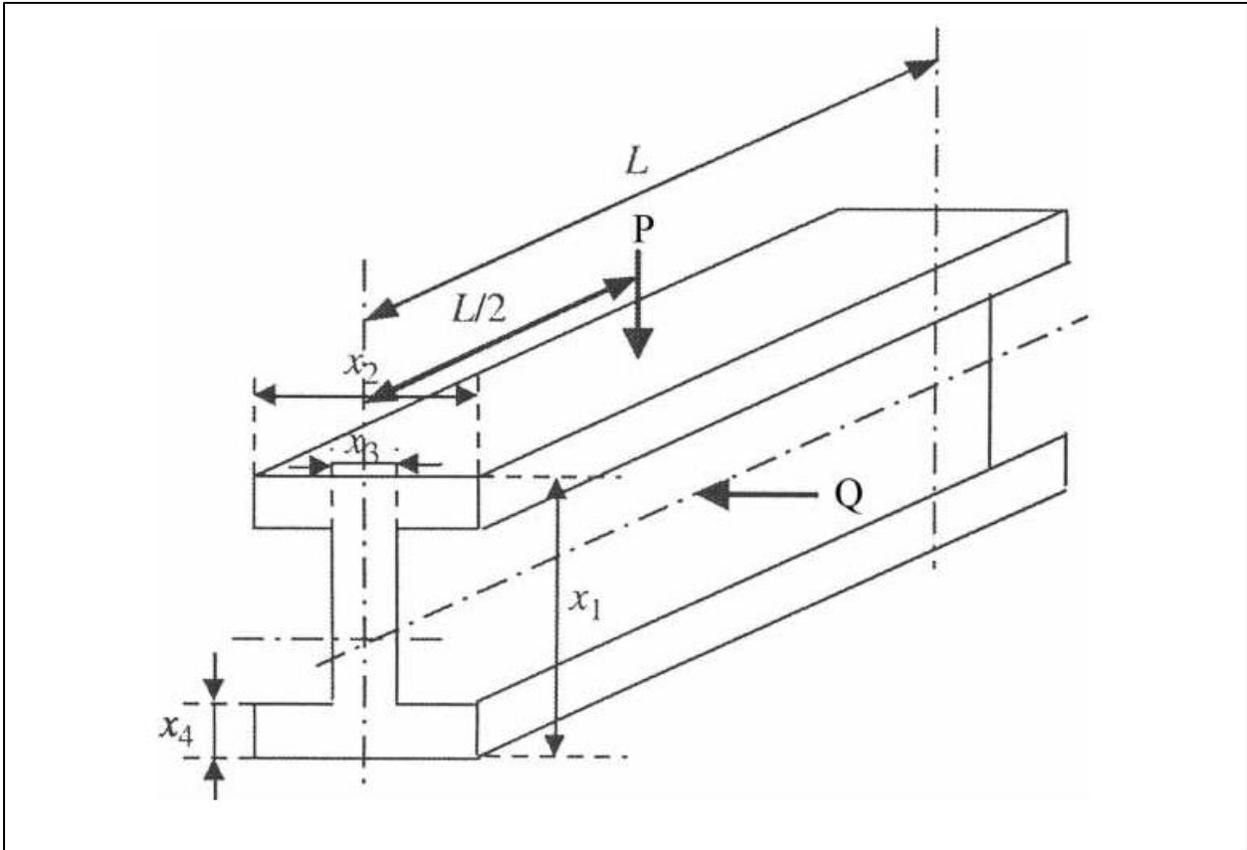


Figure 1. Physical specifications of an I-beam [5].

Geometrical limitations are stated as follows:

$$10 \leq x_1 \leq 80, 10 \leq x_2 \leq 50, 0.9 \leq x_3 \leq 5, 0.9 \leq x_4 \leq 5 \quad (12)$$

The length of I-beam is $L=200 \text{ cm}$, the external pressures are $P= 600 \text{ kN}$ and $Q=50 \text{ kN}$. Young's modulus is $E= 2 \times 10^4 \text{ kN/cm}^2$ and $\sigma_b = 16 \text{ kN/cm}^2$. Optimal solutions are shown in Tables (1), (2), and (3). The aim is to minimize $f1$ and $f2$ together. It is not possible, since they inversely relate to one another. In other words, if the amount of used material is much more, then the cross-section is thicker, which in turn, decreases bending ratio $f2$ function. When $f1$ increases, $f2$ decreases. Therefore, decreasing both functions starts a "competition" between them.

Table 1. The implementation of I-beam using the NSGAI algorithm when the population size(number of individuals) is 40 and number of generation is 50, for 10 separate run.

| | Number of Run (NSGAI) | f1 | f2 | x1 | x2 | x3 | x4 |
|---|-----------------------|-------------------|-------------------|----|-------------------|-------------------|-------------------|
| ▶ | 0 | 145.3259825746... | 0.035103599398... | 80 | 16.51880456438... | 4.528347681988... | 4.177599963721... |
| | 1 | 146.3052965538... | 0.033768217637... | 80 | 12.80192664023... | 3.634237003761... | 1.535422670671... |
| | 2 | 139.8491481304... | 0.044442357976... | 80 | 47.79389183865... | 2.543063898823... | 4.110720514092 |
| | 3 | 184.5038494667... | 0.033288883677... | 80 | 11.06736967389... | 2.923348719311... | 3.043636470075... |
| | 4 | 116.2630854503... | 0.048713613346... | 80 | 46.08418529670... | 4.634950945170... | 4.731020753705... |
| | 5 | 140.0192967048... | 0.040356326103... | 80 | 23.427599488491 | 3.845321313871... | 3.1078815040681 |
| | 6 | 201.7328397356... | 0.027573927366... | 80 | 20.57051577166... | 2.299243692727... | 4.5355701977087 |
| | 7 | 174.67444579582 | 0.032561184669... | 80 | 42.75043066253... | 4.655014071359... | 2.578074345122... |
| | 8 | 176.4495204491... | 0.040344567316... | 80 | 40.34668700320... | 3.326946531155... | 1.416499618401... |
| * | 9 | 219.1659695571... | 0.028766118026... | 80 | 18.37146154063... | 3.773292148659... | 2.484883426728... |

Table 2. The implementation of I-beam using the NSGAI algorithm when the population size(number of individuals) is 40 and number of generation is 50, for 10 separate run.

| | Number of Run (NSGAI) | f1 | f2 | x1 | x2 | x3 | x4 |
|---|-----------------------|-------------------|-------------------|----|-------------------|-------------------|-------------------|
| ▶ | 0 | 254.8875007691... | 0.016585023082... | 80 | 40.44307304101... | 0.908123039132... | 0.956240757348... |
| | 1 | 249.50932725521 | 0.016353311837... | 80 | 49.96404921634... | 2.792254290493... | 4.777627827589... |
| | 2 | 143.7463567483... | 0.034967172440... | 80 | 42.54159932608... | 0.973207667876... | 0.956240757348... |
| | 3 | 283.7628512167... | 0.014728556917... | 80 | 49.96404921634... | 1.132919980507... | 0.956240757348... |
| | 4 | 229.5480949662... | 0.018210550367... | 80 | 47.83303033459... | 0.908123039132... | 0.956240757348... |
| | 5 | 380.9756924631... | 0.010560997772... | 80 | 40.44307304101... | 0.908123039132... | 1.411317028110... |
| | 6 | 210.7701733766... | 0.021368338788... | 80 | 42.54159932608... | 1.132919980507... | 1.816154530931... |
| | 7 | 164.1370415702... | 0.031377052304... | 80 | 40.44307304101... | 0.908123039132... | 3.247144628896... |
| | 8 | 171.5427512732... | 0.028162907358... | 80 | 47.83303033459... | 0.908123039132... | 1.816154530931... |
| * | 9 | 205.1016845477... | 0.020552432314... | 80 | 40.44307304101... | 0.973207667876... | 1.816154530931... |

Table 3. The implementation of I-beam using the NAVNSGA algorithm when the population size(number of individuals) is 40 and number of generation is 50, for 10 separate run.

| | Number of Run (NAVNSGA) | f1 | f2 | x1 | x2 | x3 | x4 |
|---|-------------------------|-------------------|-------------------|--------------------|-------------------|-------------------|-------------------|
| ▶ | 0 | 151.6790396826... | 0.030750486038... | 80 | 33.10886243270... | 0.904941364286... | 3.918008759998... |
| | 1 | 192.1917837489... | 0.023166654911... | 80 | 33.10886243270... | 0.904941364286... | 3.850014191004... |
| | 2 | 143.8723217000... | 0.0332474677... | 0.0332474677464089 | 6892572... | 0.911795119853... | 1.127507337847... |
| | 3 | 151.5217454486... | 0.031499706410... | 80 | 30.540927612009 | 0.911795119853... | 1.129950159708... |
| | 4 | 171.6838861114... | 0.027574887854... | 80 | 30.31143248747... | 0.911795119853... | 1.172217635005... |
| | 5 | 234.0343328500... | 0.017727973517... | 80 | 33.10886243270... | 0.904941364286... | 3.6554063775369 |
| | 6 | 202.7652240830... | 0.021078530239... | 80 | 32.54440467084... | 0.904941364286... | 4.543563467656... |
| | 7 | 515.2389180500... | 0.007562730327... | 80 | 33.10886243270... | 0.904941364286... | 3.601708382136... |
| | 8 | 175.6488622433... | 0.025442433187... | 80 | 33.74355812638... | 0.904941364286... | 3.572931064047... |
| * | 9 | 137.3503776814... | 0.035951406785... | 80 | 33.56550928464... | 0.904941364286... | 3.278973967851... |

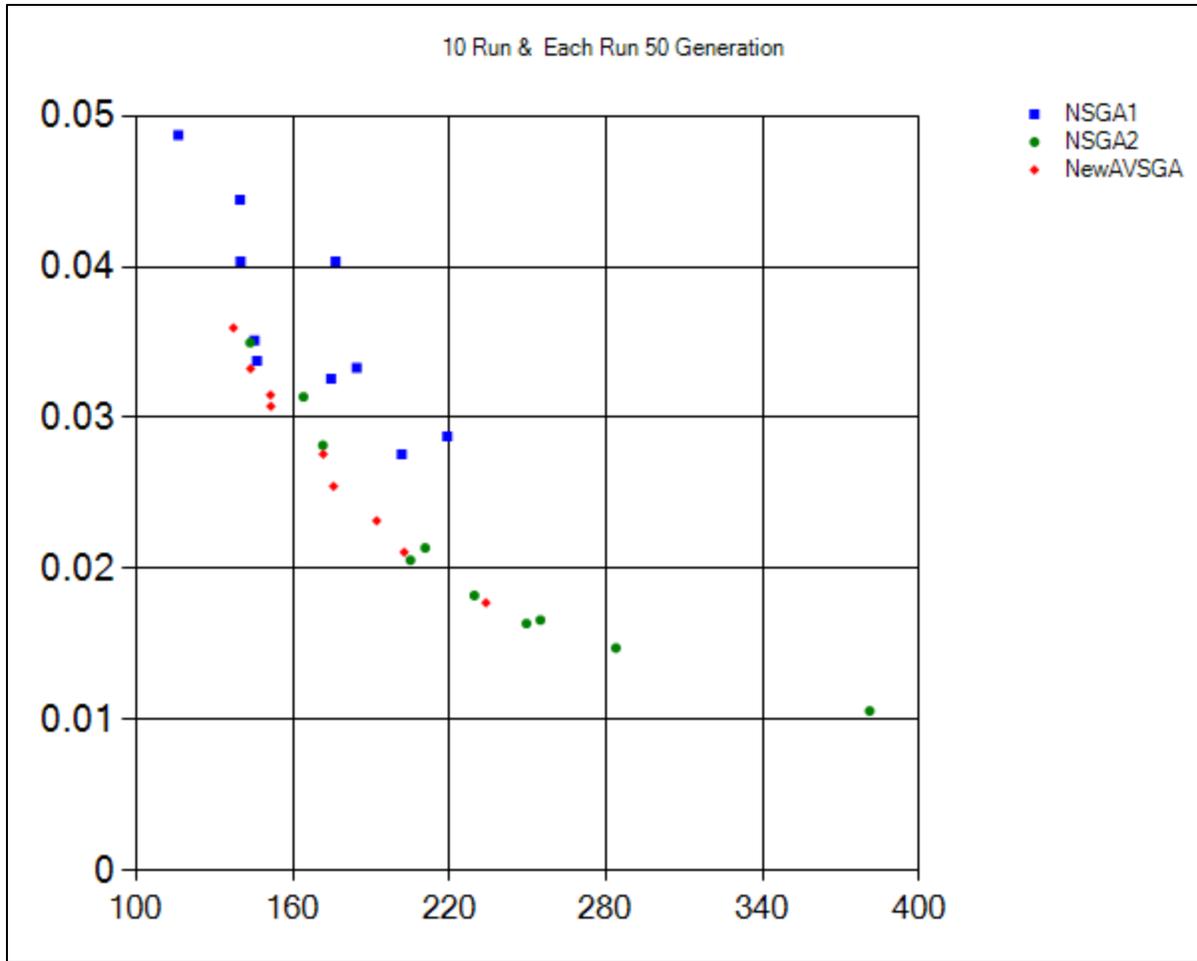


Figure 2. Comparing the mentioned algorithms in the figure above based on 10 separate run for each algorithm and number of generation is 50.

According to Fig.(3), the NAVNSGA algorithm improves the results. As can be seen, the NAVNSGA algorithm “dominate” the results obtained from two other algorithms. Results were selected based on the least f_1 among the solutions. Compared to proposed approach in [5], the proposed model highly improves the results. The program used to implement this paper is the Microsoft visual studio C# windows form.

Table 4. Results obtained from [5].

| Results obtained from | Result |
|-----------------------|-----------------------|
| $f_1(\vec{X})$ | 132.5374 |
| $f_2(\vec{X})$ | 0.0375 |
| Design variables | [80, 34.53, 0.9, 0.9] |

6. Conclusions

Multi-objective evolutionary sorting algorithms are used for solving many real-world problems, they are also widely used in engineering systems optimization. On the other hand, the increasing use of multi-objective optimization algorithms in many scientific fields is undeniable. Therefore, multi-objective optimization algorithms can solve the problems existing in many scientific problems. For further research, the NAVNSGA algorithm could be used for optimizing other forms of multi-objective problems in engineering systems and strengths and weaknesses of these algorithms are examined by future research.

Reference

- [1] Z. Falahiazar, M. Rohani and A. Falahiazar, "Controlling the False Alarm in an Intrusion Tolerant Database System Using Significance Degrees of Data Objects," *Journal of mathematics and computer science*, pp. vol. 13, pp. 212-225, 2014., 2014.
- [2] A. Falahiazar, S. Setayeshi and Y. Sharafi, "Computational model of social intelligence based on emotional learning in the Amygdala".
- [3] Y. Sharafi, S. Setayeshi and A. Falahiazar, "An Improved Model of Brain Emotional Learning Algorithm Based on Interval Knowledge," *Journal of mathematics and computer science*, 2014.
- [4] Z. Falahiazar, M. Rohani, L. Falahiazar and M. Teshnelab, "OPTIMIZING AN INTRUSION TOLERANT DATABASE SYSTEM USING NEURAL NETWORK," *IJCSNS International Journal of Computer Science and Network Security*, vol. 7.5 , pp. 224-234, 2012.
- [5] S. S. Rao and A. Kiran K., "Multi-objective optimization of engineering systems using game theory and particle swarm optimization," vol. 41.8, no. 737-752, 2009.
- [6] O. Kapliński and J. Tamošaitienė, "GAME THEORY APPLICATIONS IN CONSTRUCTION ENGINEERING AND MANAGEMENT," vol. 16(2), no. 348–363, 2010.
- [7] A. Dixit, "Thomas Schelling's contributions to game theory," vol. 108.2, no. 213-229, 2006.
- [8] H. R. Navidi, A. Amiri and R. Kamranrad, "Multi Responses Optimization Through Game Theory Approach," *Internattionall Journall off Industriall Engiineeriing & Producttiion Research*, vol. 25, no. 3, pp. 215--224, Septtember 2014.
- [9] M. Hassanpour, M. Hassanpour-ezatti, H. Navidi and T. Abachi, "A Fuzzy expert system for prescribing atorvastatin optimum dose," *Semnan University of Medical Sciences and Health Services*, vol. 13, no. 1, pp. 43-49, 2011.
- [10] D. D. Smith, "Multi-objective optimization for the multiphase design of active polymorphing wings," vol. 49.4, no. 1153-1160, 2012.
- [11] A. Nourbakhsha and S. Derakhshan, "The comparison of multi-objective particle swarm optimization and NSGA II algorithm: applications in centrifugal pumps," vol. 43, no. 10, 2011.
- [12] M. R. Nikoo , I. Varjavand , R. Kerachian, M. Pirooz and A. Karimi, "Multi-objective optimum design of double-layer perforated-wall breakwaters: Application of NSGA-II and bargaining models," vol. 47, no. 47–52, 2014.
- [13] S. Guangyong , a. Li and G. a, "Radial basis functional model for multi-objective sheet metal forming optimization," 2011.
- [14] s. s. RAO, "GAME THEORY APPROACH FOR MULTIOBJECTIVE STRUCTURAL OPTIMIZATION," vol. 25.1, no. 119-127, 21 April 1986.
- [15] S. S. Rao and . T. I. Freiheit, "A Modified Game Theory Approach to Multiobjective Optimization," vol. 113, 1991.
- [16] R. Meng, . N. Xie and . L. Wang, "Multiobjective Game Method Based on Self-Adaptive Space Division of Design Variables and Its Application to Vehicle Suspension," vol. 2014, no. Article ID 479272, 13 pages, 2014.
- [17] A. R.T. and Marler and J.S., "Survey of multi-objective optimization methods for engineering," vol. 26, no. 369–395 (2004), 2004.
- [18] M. R. Nikoo, R. Kerachian and M. H. Niksokhan, "Equitable waste load allocation in rivers using fuzzy Bi-matrix games," vol. 26.15, no. 4539-4552, 2012.
- [19] S. S. Hosseini, H. Navidi and A. Hajfathaliha, "A New Linear Programming Method for Weights Generation and Group Decision Making in the Analytic Hierarchy Process," *Group Decision and Negotiation, springer*, vol. 21, no. 3, pp. 233-254, 2012.
- [20] H. Navidia, M. Bashirib and M. Messi , "A heuristic approach on the facility layout problem based on game theory," *International Journal of Production Research, Taylor & Francis*, vol. 50, no. 6, pp. 1512-1527, 2012.
- [21] R. Ayanzadeh, A. S. Zavar Mousavi and H. Navidi, "Honey Bees Foraging Optimization for Mixed Nash Equilibrium

- Estimation," *Ayanzadeh, Ramin, Azam S. ZaHoney Bees Foraging OTrends in Applied Sciences Research*, p. 6.12, 2011.
- [22] A. Ebrahimi Zade, . A. Sadegheih and . M. M. Lotfi, "A modified NSGA-II solution for a new multi-objective hub maximal covering problem under uncertain shipments," *Journal of Industrial Engineering International*, pp. 1-13, 2014.
- [23] M. R. Nikoo, "Multi-objective optimum design of double-layer perforated-wall breakwaters: Application of NSGA-II and bargaining models," *Applied Ocean Research*, pp. 47 : 47-52., 2014.
- [24] Y.-L. LI, W. SHAO, J.-T. WANG and . H. C, "An Improved NSGA-II and its Application for Reconfigurable Pixel Antenna Design," *Radioengineering*, p. 23.2: 733, 2014.
- [25] H. Bingquan, . B. Buckley and T.-M. Kechadi, "Multi-objective feature selection by using NSGA-II for customer churn prediction in telecommunications," vol. 37.5 (2010, no. 3638-3646, 2010.
- [26] K. Deb, "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II," vol. 1917, no. 849-858, 2000.
- [27] K. Deb and A. R. Reddy, "Reliable classification of two-class cancer data using evolutionary algorithms," vol. 72(1–2), no. 111–129, 2003.
- [28] M. Sefrioui and J. Perlaux, "Nash genetic algorithms: examples and applications," Vols. on. Vol. 1. IEEE, 2000., 2000.
- [29] L. Falahiazar, M. Teshnehlab and A. Falahiaza, "Parallel genetic algorithm based on a new migration strategy," in *Recent Advances in Computing and Software Systems (RACSS), 2012 International Conference on*, 2012.
- [30] A. Rostamian, M. Hosseinzadeh and A. Shokrollahi, "Transmission Loss Allocation in the Deregulated Electricity Market based on the Cooperative Game Theory," *The Journal of Mathematics and Computer Science*, vol. 4 , no. 1, pp. 81 - 92, 2012.
- [31] M. Khanjary and H. R. Navidi, "Optimizing Traffic Light of an Intersection by using Game Theory," *AWERProcedia Information Technology & Computer Science*, vol. 03, pp. 1163-1168, 2013.
- [32] B. Huang, B. Buckley and T. Kechadi, "Multi-objective feature selection by using NSGA-II for customer churn prediction in telecommunications," vol. 37, no. 3638–3646, 2010.
- [33] X. D. Wang, "Multi-objective optimization of turbomachinery using improved NSGA-II and approximation model," *Computer Methods in Applied Mechanics and Engineering*, pp. Wang, X. D., et al. "Multi-objective optimization of turbomachinery using improved NSGA-II and a200.9: 883-895, (2011.
- [34] A. Rostamian, M. Hosseinzadeh and A. Shokrollahi, "Transmission Loss Allocation in the Deregulated Electricity Market based on the Cooperative Game Theory," *The Journal of Mathematics and Computer Science*, pp. Vol. 4 No.1 (2012) 81 - 92, 2012.