Turkish Journal of Science & Technology
17(2), 223-232, 2022

Research Paper
https://doi.org/10.55525/tjst.1160814

Performance Analysis of Current Multi-Objective Metaheuristic Optimization Algorithms for Unconstrained Problems

Eyüp ERÖZ[1*], Erkan TANYILDIZI[2]
[1] Software Engineering, Technology Faculty, Firat University, Elazig, Turkey
[2] Software Engineering, Technology Faculty, Firat University, Elazig, Turkey
[*1] eeroz@firat.edu.tr, [2] etanyildizi@firat.edu.tr

(Geliş/Received: 11/08/2022; Kabul/Accepted: 13/09/2022)

Abstract: Multi-objective optimization is a method used to produce suitable solutions for problems with more than one Objective. Various multi-objective optimization algorithms have been developed to apply this method to problems. In multi-objective optimization algorithms, the pareto optimal method is used to find the appropriate solution set over the problems. In the Pareto optimal method, the Pareto optimal set, which consists of the solutions reached by the multi-objective optimization, includes all the best solutions of the problems in certain intervals. For this reason, the Pareto optimal method is a very effective method to find the closest value to the optimum. In this study, the Multi-Objective Golden Sine Algorithm we developed (MOGoldSA), the recently published Multi-Objective Artificial Hummingbird Algorithm (MOAHA), and the Non-Dominant Sequencing Genetic Algorithm II (NSGA-II), which has an important place among the multi-objective optimization algorithms in the literature, are discussed. In order to see the performance of the algorithms on unconstrained comparison functions and engineering problems, performance comparisons were made on performance metrics

Key words: Multi-objective optimization, Pareto Optimal, unconstrained benchmark functions, performance metrics

Güncel Çok Amaçlı Metasezgisel Optimizasyon Algoritmalarının Kısıtsız Problemler için Performans Analizi

Öz: Çok amaçlı optimizasyon, birden fazla amacı bulunan problemlere uygun çözümler üretmek için kullanılan bir yöntemdir. Bu yöntemi problemlere uygulamak amacıyla çeşitli çok amaçlı optimizasyon algoritmaları geliştirilmiştir. Çok amaçlı optimizasyon algoritmalarında problemler üzerinden uygun çözüm kümesi bulmak için pareto optimal yöntemi kullanılmıştır. Pareto optimal yönteminde, çok amaçlı optimizasyonun ulaştığı çözümlerden oluşan pareto optimal kümesi, problemlerin belli aralıklardaki tüm en iyi çözümlerini içermektedir. Bu nedenle pareto optimal yöntemi, optimuma en yakın değeri bulmak için oldukça etkili bir yöntemdir. Bu çalışmada, geliştirdiğimiz Çok Amaçlı Altın Sinüs Algoritmasının (MOGoldSA), son zamanlarda yayınlanan Çok Amaçlı Yapay Sinekkuşu Algoritması (MOAHA) ve literatürde çok amaçlı optimizasyon algoritmaları içerisinde önemli yere sahip Baskın Olmayan Sıralama Genetik Algoritması II (NSGA-II) ele alınmıştır. Algoritmaların kısıtsız kıyaslama fonksiyonları ve mühendislik problemleri üzerindeki başarımını görmek için performans metrikleri üzerinde performans karşılaştırılması yapılmıştır.

Anahtar kelimeler: Çok amaçlı optimizasyon, Pareto Optimal, kısıtsız kıyaslama fonksiyonları, performans metrikleri

1. Introduction

From past to present, people have had to struggle with many problems. Problems have become more complex and difficult to solve with classical (stochastic and deterministic) methods. For this reason, the importance of modeling, calculation and algorithm studies carried out by experts on solving problems is increasing [1].

The heuristic optimization method is one of the leading methods developed to solve problems. Optimization, which is the process of producing suitable solutions in line with the objective of the problem, has an important place for the solution of problems in many areas encountered in daily life. Vehicle routing, logistics, engineering, business plan, mapping, etc. fields and engineering and benchmarking functions are examples of places where optimization is effective [2].

Various optimization algorithms have been developed for the implementation of the optimization method. In order for algorithms to produce solutions on the problem, they need to create a mathematical model for the Objective. Since mathematical modeling in complex problems is very costly and difficult, algorithms that need

[*] Corresponding author: eeroz@firat.edu.tr. ORCID Number of authors: [1] 0000-0003-2670-0606, [2] 0000-0003-2973-9389

modeling are insufficient to produce solutions. Since the same is true for most linear optimization models, metaheuristic methods have been used.

Since metaheuristic methods examine the search space efficiently and actively regardless of the problem, they find the most appropriate solutions to the problems that deterministic methods cannot solve in a reasonable time. Metaheuristic methods do not always find the global optimum, but they are very useful because they produce the most effective solution to the global optimum in the fastest time [3,4]. In general, metaheuristic methods are evaluated in seven different categories: physics, mathematics, chemistry, biology, social, music and herd-based. These algorithms produce solutions by mimicking ethological, biological and physical behaviors [5].

Multi-objective optimization algorithms are being developed to produce effective solutions to problems with multiple Objectives encountered in daily life. Examples of these algorithms are the Arrow Dragonfly Algorithm (MODA)[6], Multi-Objective Ant Lion Optimization (MOALO)[7], Multi-Objective Differential Evolution (MODE)[8], Multi-Objective Gray Wolf Optimization (MOGWO)[9], Algorithms such as Multi-Objective Particle Swarm Optimization (MOPSO)[10], Non-Dominant Sequence Genetic Algorithm II (NSGA)[11] and Multi-Objective Artificial Hummingbird Algorithm (MOAHA)[12] can be given as examples of current multi-objective optimization algorithms[13].

In this study, the effects of the MOGoldSA algorithm we developed, the newly developed MOAHA and the NSGA-II metaheuristic multi-objective optimization algorithms, which have an important place in the literature, on unconstrained comparison functions and engineering problems are examined through success criteria.

2. Performance Comparison Optimization Algorithms

2.1 Multi-Objective Golden Sine Algorithms (MOGoldSA)

It is the version of Gold-SA developed for single-objective optimization algorithms, which has been made applicable to problems with more than one purpose. It is a mathematics-based metaheuristic optimization algorithm developed based on the sine function. The sine function can be defined as the coordinate of a point to the y-axis on a 1-unit radius circle whose center is the origin. It is calculated using the angle that the line drawn from the origin to the point makes with the y-axis. Since the values of the sine function with a definition range of [-1,1] are repeated at regular intervals, it is characterized as a periodic function [14].

Scanning all values of the sine function in the unit circle is similar to scanning the search space in optimization problems. Based on this similarity, Gold-SA was developed. Like all other swarm-based optimization algorithms, Gold-SA starts with a randomly generated population. For population-based algorithms, it is very important to choose the first population well. As shown in Equation 1, the Gold-SA initial population aims to better scan the search space by generating a random distribution for each dimension.

$$V = \mathrm{rand}(\mathrm{agent\_no}, \mathrm{size}) * (\mathrm{ub} - \mathrm{lb}) + \mathrm{lb} \tag{1}$$

The main purpose of metaheuristic methods is to search for the regions considered to be the best in the search area and to make sure that these areas are scanned as much as possible. Gold-SA uses the gold section method to do this as best as possible. The MOGoldSA method has been introduced in order to apply Gold-SA to multi-purpose problems due to its important features such as wide scanning of the search space, producing near-optimal results and fast working while producing solutions to problems. The pseudo-code of MOGoldSA is given in Figure 2 [12].

Calculate the initial population by Equation 2.1.
generate randomly based on uniform distribution as the number of search agents for each dimension
Calculate availability of search agents
Assign top search agent as target
Create archive
while maximum iteration
  Get function values
  Find the best search agents and update Archive with Roulette wheel
  if archive is full
    Run the archive maintenance mechanism to remove one of the existing archive members
    Update archive by adding new solutions to archive
  else
    Update archive by adding new solutions to archive
  end if
  for search agent no
    $r \leftarrow rand(0,1)$
    $r_1 \leftarrow 2\pi * r$
    $r_2 \leftarrow \pi * r$
    for size number
      $V(i,j) = V(i,j) * |\sin r_1| - r_2 * \sin r_1 * |x_1 * D(j) - x_2 * V(i,j)|$
    end for
  end for
Find the best solution (search agent) and assign D(j) as target value
  if $V(i,j) < D(j)$
  then $b \leftarrow x_2, x_2 = x_1$
    $x_1 \leftarrow a * \varphi + b * (1 - \varphi)$
  else $a \leftarrow x_1, x_1 = x_2$
    $x_2 \leftarrow a * (1 - \varphi) + b * \varphi$
  if $x_1 == x_2$
  then $a = random1, b = random2$
    $x_1 = a * \varphi + b * (1 - \varphi)$
    $x_2 = a * (1 - \varphi) + b * \varphi$
  end if
end while
return The best solution set and the global optimum result obtained

Figure 2.1 Pseudo code of MOGoldSA

2.1 Multi-Objective Artificial Hummingbird Algorithms (MOAHA)

MOAHA is an algorithm developed by Artificial Hummingbird Algorithm (AHA) for multi-objective optimization problems[15]. MOAHA starts with a random solution set and a random artificial hummingbird population by creating a fixed number of external archives. All non-dominant solutions are archived and the visit table is initialized. The visitation chart monitors the level of hummingbird visitation of each food source and is a very important component of the MOAHA algorithm. When the food source is not visited by the hummingbird for a long time, the value in the visitation table is high. In this case, the priority of visiting hummingbirds will be increased as the food source is more voluminous. During each iteration, the MOAHA has a 50% probability of performing a guided foraging or regional foraging. During the guided search, each hummingbird updates the location for the selected target food source and the dominance relationship for the visit table. In regional foraging, updates are made according to the local population. As a result of the search, an NDS-based solution update is applied and the visit table is updated. In every 2n iterations, a migration search is performed and the solutions on the worst front are randomly started and the visit table is changed. At the end of the iteration, if the non-dominant solutions in the new population exceed the size, they are added to the archive according to the external archiving procedure based on DECD. All these processes are repeated until the maximum number of iterations is reached. Finally, the archive with the optimal solution set is returned [12].

Non-Dominant Ranking (NDS); The basic principle is that solutions dominated by fewer solutions have a higher dominance hierarchy. All solutions are ranked according to their dominance level. There are three different solution update states in NDS. These situations are shown in Figure 2.1.

(1) If the candidate solution front is better than the current solution front, the candidate solution replaces the current solution
(2) If the candidate solution and the current solution front are equal, the probability of being selected is 50%.
(3) If the candidate solution front is worse than the current solution front, there is no change and the next iteration continues.

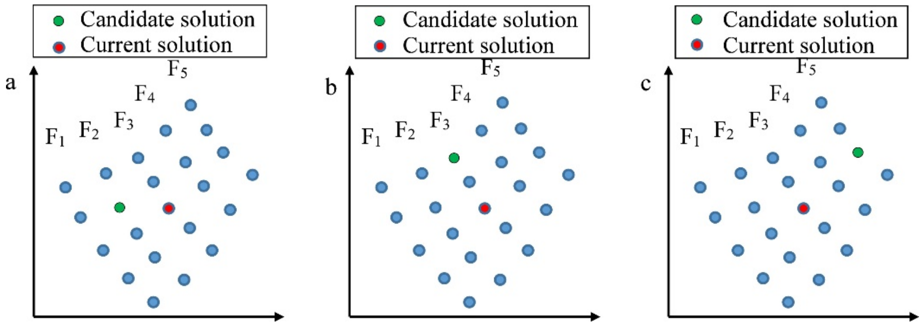

Figure 2.2 a) The front of the candidate solution is better than the front of the current solution, b) The probability of choosing between the candidate and existing solutions is 50%, c) The front of the current solution is better than the front of the candidate solution [12]

DECD method; The crowd distance method is an effective parameterless method to increase the variety of solutions [16]. For this reason, it is used in most multi-objective optimization algorithms to maintain a fixed size external archive by removing excessive solutions with smaller crowd distance. When a solution is removed, the order of the crowd distance of the remaining solutions will change. Therefore, the next solution to be removed in terms of the initial crowd order may not have the current crowd order, reducing the variety of solutions in the archive. The crowd-distance-based archiving procedure is ineffective in maintaining the sustainable diversity of the optimal solution set. Therefore, an external archive with DECD is recommended. In the DECD method, when the solution with the minimum crowd distance present in the optimal solution set is removed, only the crowd distances of the solutions adjacent to the removed solution need to be updated and the crowd distances of the remaining solutions do not change. To generate n solutions to create an external archive using the DECD method, the crowd distances of the 2n solutions need to be recalculated.

While adding multiple strategies, the theoretical analysis is as follows;
(1) An external archive is created to record non-dominant optimal solutions. Using the archive for storage can greatly benefit all multi-objective optimization algorithms[17]. The DECD method is used to manage the solutions in the archive. It has been proven in the literature that the translation distance is suitable for maintaining solution diversity. A phase out strategy can significantly improve solution delivery [18]. Therefore, DECD can benefit uniformity and solution diversity.

(2) The NDS method can count the distribution of all solutions on the fronts that can be easily compared while performing the sorting [16]. Thus, better non-dominant solutions may be kept in the archive. It can be passed to the next iteration to allow other individuals to search. In other words, NDS can facilitate the algorithm to reach optimal solutions.

2.3 Non-Dominant Sorting Genetic Algorithm II (NSGA-II)

Non-Dominant Sorting Genetic Algorithm II (NSGA-II) is a multi-objective genetic algorithm proposed by Deb et al. in 2002. It is an extension and development of the NSGA previously proposed by Srinivas and Deb in 1995. In the structure of NSGA-II, in addition to genetic operators, crossover and mutation, two special multi-objective operators and mechanisms have been defined and used.

Non-dominated Sorting: Population is sorted and sorted by F1, F2, etc. is partitioned. Here F1 (first front part) shows the approximate Pareto front. Crowding Distance: It is a sorting mechanism between members of a front where each other dominates or dominates. These sequencing mechanisms are used in conjunction with genetic selection operators (usually Tournament Selection Operator) to create the next generation population [11]. The pseudo code of NSGA II is given in Table 4.3.

```
Initialize population: P
Generate random population size: N
Evaluate Goal Values
Assign Sort (level) by Pareto order
Create Child Population
      Binary Tournament Selection
      Recombination and Mutation
For fi = 1 to g do
       For Every Parent and Child of the Population
              Sort (level) assignment according to Pareto order
              Creates sets of non-dominant solutions
              Determine crowd distance
              Cycle adding new solutions to the next generation, from the first front to N
individuals
        end
Choose spots on lower levels with high crowd distance
create next generation
      Binary Tournament Selection
      Recombination and Mutation
end
```

Figure 2.3 Psuedo code of NSGA-II

3. Experiments and Results

In order to test the success of the developed method, the performance values of 11 different unconstrained comparison functions (unconstrained function and engineering problems) in the literature were examined. The mathematical expressions of some of the functions used are given in Table 3.1.

Table 3.1 Mathematical representation of the Unconstrained functions used

F	LIMIT OF VARIABLE	OBJECTIVE FUNCTIONS
ZDT1	$x_i \in [0,1]$ $i = 1,\ldots,n$ $n = 30$	$f_1(x) = x_1$ $f_2(x) = g(x)\left(-\sqrt{f_1/g(x)}\right)$ $g(x) = 1 + 9\left(\sum_{i=2}^{m} x_i\right)\Big/(n-1)$
ZDT6	$x_i \in [0,1]$ $i = 1,\ldots,n$ $n = 10$	$f_1(x) = 1 - \exp(-4x_1)sin^6(\pi x_1)$ $f_2(x) = g(x)(1 - (x_1/g(x))^2)$ $g(x) = 1 + 9\left[\left(\sum_{i=2}^{m} x_i/(n-1)\right)\right]^{0.25}$
FON	$x_i \in [-4,4]$ $i = 1,\ldots,n$ $n = 10$	$f_1(x) = 1 - exp\left(-\sum_{1}^{3}\left(x_i - \frac{1}{\sqrt{3}}\right)^2\right)$

Table 3.1 Mathematical representation of the Unconstrained functions used (continued)

F	LIMIT OF VARIABLE	OBJECTIVE FUNCTIONS
KUR	$x_i \in [-5,5]$ $i = 1, \dots, n$ $n = 3$	$f_1(x) = \sum_{i=1}^{n-1}\left(-10exp\left(-0.2\sqrt{x_i^2} + \sqrt{x_{i+1}^2}\right)\right)$ $f_2(x) = \sum_{i=1}^{n}\left(\lvert x_i\rvert^{0.8} + 5\sin x_i^3\right)$
POLO	$x_i \in [-\pi, \pi]$ $i = 1, \dots, n$ $n = 2$	$f_1(x) = [1 + (A_1 - B_1)^2 + (A_2 - B_2)^2]$ $f_2(x) = [(x_1 + 3)^2 + (x_2 + 1)^2]$ $A_1 = 0.5sin1 - 2cos1 + sin2 - 1.5cos2$ $A_2 = 1.5sin1 - cos1 + 2sin2 - 0.5cos2$ $B_1 = 0.5sinx_1 - 2cosx_1 + sinx_2 - 1.5cosx_2$ $B_2 = 1.5sinx_1 - cosx_1 + 2sinx_2 - 0.5cosx_2$
FBTP	$1 \le x_1, x_4 \le 3$ $\sqrt{2} \le x_2, x_3 \le 3$ F=10, E=2e5, L= 200	$minf_1(x) = L(2x_1 + \sqrt{2x_2} + \sqrt{x_3} + x_4)$ $minf_2(x) = (F\frac{L}{E})(\frac{2}{x_2} + 2\frac{\sqrt{2}}{x_2} - 2\frac{\sqrt{2}}{x_3} + \frac{2}{x_4})$
GEAR	$x_i \in [12,60]$ $i = 1, \dots, n$ $n = 4$	$f_1(x) = \left(\left(\frac{1}{6.931}\right) - \left(\frac{x_1x_2}{x_3x_4}\right)\right)^2$ $f_2(x) = \max([x_1, x_2, x_3, x_4])$

The criteria used to evaluate the success of the developed algorithms are called performance criteria. In our study, performance criteria, which have taken place in the literature and are frequently used in the comparison of multi-objective optimization algorithms, are used. These; Distance Distance (SE)[19], Inverse Distance Distance (RGD)[19,20], Space(S)[21], Spread (SP)[22], Maximum Spread(MS)[22,23] and High Volume Indicator (HV)[19] are performance measurement metrics. The explanations of the metrics are given in Table 3.2.

Table 3.2 Performance Criteria Used

Metric Name	Criteria	Definition
GD	Convergence	The lower the GD value, the better for convergence.
RGD	Convergence	The lower the RGD value, the better for convergence.
S	Variation	The lower the S value, the better for convergence.
SP	Spread	The higher the SP value, the better for propagation.
MS	Spread	The higher the MS value, the better for propagation.
HV	Convergence and Diversity	Higher HV value is better for all criteria

When Table 3.3 is looked at, statistical data are seen according to GD and RGD criteria. According to the GD criterion, MOGoldSA achieved 9/11 and MOAHA 2/11 success out of 11 comparison functions. When compared according to the RGD criterion, the MOAHA algorithm was successful at a rate of 9/11, and the MOGoldSA was successful at a rate of 2/11. In this case, the superiority of MOGoldSA according to the GD success metric and MOAHA according to the RGD criterion is clearly seen.

Looking at Table 3.4, statistical data according to MS and S criteria can be seen. In these criteria, as seen in Table 3.2, higher for MS and lower for S means better. Accordingly, MOGoldSA 4/11 and NSGA-II 7/11 were successful for the MS criterion. In the S criterion, 4/11 MOGoldSA, 6/11 MOAHA and 1/11 NSGA-II algorithms were successful.

Table 3.3 Statistical data according to GD and RGD criteria

F	I	MOGOLDSA	MOAHA	NSGA-II	MOGOLDSA	MOAHA	NSGA-II
		GD			RGD		
FBTP	Best	**1,3538**	22,2135	1,522E+04	**4,8131**	100,4478	1,344E+04
	Mean	1,7139	22,2268	1,724E+06	5,3122	113,8351	4,596E+04
	Worst	2,2770	22,2396	3,301E+07	6,0131	127,9738	2,770E+05
	Standart	0,2163	0,0059	7,365E+06	0,3271	0,0855	6,400E+04
	Time	394,6623	207,6105	4236,8051	394,6623	207,6105	4236,8051
GEAR	Best	4,852E+04	**519,4864**	1,371E+10	5,431E+04	**364,8572**	1,746E+07
	Mean	1,179E+05	573,0392	3,826E+10	7,558E+04	402,6668	5,789E+09
	Worst	3,278E+05	626,3775	1,324E+11	1,368E+05	489,9123	3,884E+10
	Standart	6,615E+04	266,1139	3,176E+10	2,339E+04	285,6714	1,052E+10
	Time	299,9048	215,4202	4523,8692	299,9048	215,4202	4523,8692
FON	Best	**0,0010**	0,0046	0,0022	0,0056	**0,0014**	0,0046
	Mean	0,0013	0,0049	0,0025	0,0062	0,0018	0,0049
	Worst	0,0018	0,0051	0,0030	0,0076	0,0020	0,0057
	Standart	0,0002	0,0001	0,0003	0,0004	0,0001	0,0003
	Time	430,5992	167,6028	4338,8464	430,5992	167,6028	4338,8464
KUR	Best	16,0989	15,6197	16,1477	**15,5874**	16,1614	15,6148
	Mean	16,1460	**15,6336**	16,1936	15,5995	16,1736	15,6299
	Worst	16,1950	15,6388	16,2691	15,6140	16,1922	15,6399
	Standart	0,0289	0,0047	0,0270	0,0060	0,0086	0,0077
	Time	213,2565	131,9476	4406,5289	213,2565	131,9476	4406,5289
POL	Best	**0,0303**	0,0687	0,3466	0,1083	**0,0216**	1,2796
	Mean	0,2755	0,0745	0,4699	0,1468	0,0566	3,0514
	Worst	0,5155	0,0827	0,6717	0,1838	0,0876	9,6327
	Standart	0,1651	0,0036	0,1122	0,0234	0,0243	3,3753
	Time	224,9433	189,4781	4294,0028	224,9433	189,4781	4294,0028
VIE	Best	**0,0067**	0,0506	16,2128	0,0472	**0,0017**	0,1080
	Mean	0,0124	0,0550	18,5965	0,0711	0,0020	0,1323
	Worst	0,0221	0,0611	19,8853	0,1124	0,0024	0,1779
	Standart	0,0048	0,0028	0,9337	0,0138	0,0002	0,0175
	Time	545,3776	183,1814	4870,8179	545,3776	183,1814	4870,8179
ZDT1	Best	**0,0007**	0,0046	7,6003	0,0065	**0,0012**	6,0681
	Mean	0,0017	0,0049	8,8265	0,0076	0,0016	6,9284
	Worst	0,0038	0,0052	9,6329	0,0103	0,0022	7,6887
	Standart	0,0007	0,0002	0,6086	0,0010	0,0003	0,4094
	Time	662,2493	141,3644	4053,5786	662,2493	141,3644	4053,5786
ZDT2	Best	**0,0004**	0,0047	6,223E+07	0,0059	**0,0005**	1,757E+03
	Mean	0,0005	0,0049	1,508E+11	0,0073	0,0012	4,575E+05
	Worst	0,0006	0,0053	2,645E+12	0,0083	0,0022	7,931E+06
	Standart	0,0000	0,0002	5,921E+11	0,0005	0,0004	1,762E+06
	Time	702,9784	157,1703	4786,8025	702,9784	157,1703	4786,8025
ZDT3	Best	**0,0011**	0,0052	1,4137	0,0068	**0,0009**	1,8811
	Mean	0,0015	0,0053	7,7594	0,0086	0,0011	7,1750
	Worst	0,0024	0,0056	12,3287	0,0144	0,0012	12,6277
	Standart	0,0003	0,0097	3,2963	0,0016	0,0008	3,0546
	Time	442,1689	142,1434	4218,0993	442,1689	142,1434	4218,0993
ZDT4	Best	**0,0021**	0,0047	1,5646	0,0070	**0,0010**	0,7822
	Mean	0,0036	0,0048	4,2266	0,0090	0,0013	3,5352
	Worst	0,0061	0,0050	17,4885	0,0112	0,0017	17,6962
	Standart	0,0010	0,0074	5,4102	0,0012	0,0002	5,7539
	Time	347,9827	152,1434	4269,7994	347,9827	152,1434	4269,7994
ZDT6	Best	**0,0003**	0,0036	7,877E+36	0,0055	**0,0003**	7,878E+36
	Mean	0,0003	0,0036	1,449E+44	0,0068	0,0402	1,449E+44
	Worst	0,0004	0,0038	1,408E+45	0,0091	0,1488	1,408E+45
	Standart	0,0000	0,0002	4,323E+44	0,0009	0,0408	4,323E+44
	Time	683,4443	207,2610	5032,5692	683,4443	207,2610	5032,5692

Table 3.4 Statistical data according to MS and S criteria

F	I	MOGOLDSA	MOAHA	NSGA-II	MOGOLDSA	MOAHA	NSGA-II
		MS			S		
FBTP	Best	**545,7294**	0,6163	8,839E+02	**78,7184**	80,5914	2,442E+02
	Mean	551,7484	0,5908	1,735E+06	134,2362	121,2974	2,897E+05
	Worst	553,3824	0,5627	3,380E+07	176,2839	157,2897	5,550E+06
	Standart	1,9343	0,0146	7,547E+06	26,9925	17,3091	1,238E+06
	Time	394,6623	207,6105	4236,8051	394,6623	207,6105	4236,8051
GEAR	Best	1,021E+07	0,4672	**3,283E+10**	1,964E+06	**205,1385**	1,746E+09
	Mean	1,167E+07	0,3276	6,638E+10	2,628E+06	279,4360	8,706E+09
	Worst	1,282E+07	0,2764	1,271E+11	3,357E+06	317,7173	3,110E+10
	Standart	8,638E+05	0,0469	2,844E+10	3,650E+05	299,9545	7,029E+09
	Time	299,9048	215,4202	4523,8692	299,9048	215,4202	4523,8692
FON	Best	**1,3732**	0,1784	0,0022	0,0785	**0,0626**	0,0747
	Mean	1,3832	0,1411	0,0025	0,0939	0,0826	0,0981
	Worst	1,3884	0,1149	0,0030	0,1144	0,0970	0,1124
	Standart	0,0054	0,0152	0,0003	0,0087	0,0085	0,0097
	Time	430,5992	167,6028	4338,8464	430,5992	167,6028	4338,8464
KUR	Best	12,6658	0,8029	**16,1477**	1,2163	**1,1087**	1,8929
	Mean	12,8601	0,7923	16,1936	1,6905	1,5875	2,1941
	Worst	12,8978	0,7842	16,2691	2,1380	2,0207	2,6364
	Standart	0,0541	0,0050	0,0270	0,2524	0,2989	0,1932
	Time	213,2565	131,9476	4406,5289	213,2565	131,9476	4406,5289
POL	Best	**29,2133**	0,4224	0,3466	**2,6133**	3,0190	3,5888
	Mean	31,4311	0,2931	0,4699	4,7876	3,7678	4,5154
	Worst	32,8554	0,1967	0,6717	6,2262	5,0254	5,5066
	Standart	1,2887	0,0863	0,1122	0,9793	0,5071	0,4832
	Time	224,9433	189,4781	4294,0028	224,9433	189,4781	4294,0028
VIE	Best	8,0552	0,7767	**16,2128**	**1,4520**	1,7465	15,8646
	Mean	8,3637	0,7223	18,5965	1,9821	2,0917	20,4540
	Worst	8,4388	0,6566	19,8853	2,6379	2,5596	25,8887
	Standart	0,0800	0,0349	0,9337	0,3704	0,2333	2,4413
	Time	545,3776	183,1814	4870,8179	545,3776	183,1814	4870,8179
ZDT1	Best	1,3606	0,1841	**7,6003**	0,0559	**0,0505**	0,1332
	Mean	1,3917	0,1522	8,8265	0,0717	0,0697	0,6825
	Worst	1,4088	0,1210	9,6329	0,0912	0,0882	0,9233
	Standart	0,0125	0,0161	0,6086	0,0100	0,0089	0,2263
	Time	662,2493	141,3644	4053,5786	662,2493	141,3644	4053,5786
ZDT2	Best	1,3643	0,1764	**6,223E+07**	0,0625	**0,0568**	2,457E+06
	Mean	1,3941	0,1517	1,508E+11	0,0769	0,0749	1,324E+11
	Worst	1,4111	0,1140	2,645E+12	0,0883	0,0955	2,621E+12
	Standart	0,0152	0,0158	5,921E+11	0,0061	0,0089	5,857E+11
	Time	702,9784	157,1703	4786,8025	702,9784	157,1703	4786,8025
ZDT3	Best	**1,9235**	0,2551	1,4137	0,1825	0,1515	**0,0000**
	Mean	1,9466	0,2186	7,7594	0,2211	0,1946	1,5494
	Worst	1,9648	0,1931	12,3287	0,2711	0,2371	7,4420
	Standart	0,0122	0,0192	3,2963	0,0264	0,0236	1,8608
	Time	442,1689	142,1434	4218,0993	442,1689	142,1434	4218,0993
ZDT4	Best	1,3938	0,1821	**1,5646**	**0,0535**	0,0576	0,2266
	Mean	1,4117	0,1609	4,2266	0,0690	0,0695	0,5705
	Worst	1,4141	0,1403	17,4885	0,0886	0,0810	1,0606
	Standart	0,0046	0,0136	5,4102	0,0092	0,0063	0,2767
	Time	347,9827	152,1434	4269,7994	347,9827	152,1434	4269,7994
ZDT6	Best	1,1466	1,7003	**7,877E+36**	0,0697	**0,0443**	3,938E+21
	Mean	1,1670	0,9235	1,449E+44	0,0864	0,2547	3,662E+40
	Worst	1,1687	0,1086	1,408E+45	0,1085	0,7843	4,127E+41
	Standart	0,0049	0,7189	4,323E+44	0,0113	0,2629	1,034E+41
	Time	683,4443	207,2610	5032,5692	683,4443	207,2610	5032,5692

A high value in these criteria indicates the ideal solution. While NSGA-II 6/11, MOAHA 3/11 and MOGoldSA 2/11 were successful according to SP criteria, MOAHA 4/11 and NSGA-II 7/11 success rates were statistically successful according to HV criteria.

Table 3.5 Statistical results according to SP and HV criteria

F	I	MOGOLDSA	MOAHA	NSGA-II	MOGOLDSA	MOAHA	NSGA-II
		SP			HV		
FBTP	Best	0,8277	**1,0506**	0,9921	0,0000	0,0000	**1,0000**
	Mean	0,9161	1,0506	1,1328	0,0000	0,0000	1,0000
	Worst	0,9899	1,0506	1,4065	0,0000	0,0000	1,0000
	Standart	0,0525	0,0456	0,1524	0,0000	0,0000	0,0000
	Time	394,6623	207,6105	4236,8051	394,6623	207,6105	4236,8051
GEAR	Best	0,7453	0,7890	**1,1597**	0,8910	0,9040	**0,9920**
	Mean	0,8089	0,7936	1,3023	0,6582	0,6859	0,2377
	Worst	0,8954	0,8023	1,3753	0,0000	0,0000	0,0000
	Standart	0,0445	0,0024	0,0628	0,2848	0,3312	0,3811
	Time	299,9048	215,4202	4523,8692	299,9048	215,4202	4523,8692
FON	Best	**0,6229**	0,3619	0,2776	**0,0785**	0,0040	0,0747
	Mean	0,7041	0,3623	0,3284	0,0939	0,0009	0,0981
	Worst	0,7831	0,3626	0,3669	0,1144	0,0000	0,1124
	Standart	0,0491	0,0002	0,0293	0,0087	0,0010	0,0097
	Time	430,5992	167,6028	4338,8464	430,5992	167,6028	4338,8464
KUR	Best	0,8811	**0,9901**	0,8316	1,2163	1,0000	**1,8929**
	Mean	0,9085	0,9901	0,8418	1,6905	0,8983	2,1941
	Worst	0,9276	0,9901	0,8590	2,1380	0,8550	2,6364
	Standart	0,0156	0,0005	0,0069	0,2524	0,0465	0,1932
	Time	213,2565	131,9476	4406,5289	213,2565	131,9476	4406,5289
POL	Best	**1,0821**	1,0812	0,7580	2,6133	0,7750	**3,5888**
	Mean	1,1438	1,0832	1,0109	4,7876	0,1133	4,5154
	Worst	1,1919	1,0884	1,1176	6,2262	0,0000	5,5066
	Standart	0,0320	0,0030	0,1217	0,9793	0,2768	0,4832
	Time	224,9433	189,4781	4294,0028	224,9433	189,4781	4294,0028
VIE	Best	0,7333	0,8686	**0,9875**	1,4520	1,0000	**15,8646**
	Mean	0,8610	0,8686	1,0218	1,9821	0,6671	20,4540
	Worst	0,9945	0,8686	1,0582	2,6379	0,4160	25,8887
	Standart	0,0654	0,0000	0,0186	0,3704	0,2061	2,4413
	Time	545,3776	183,1814	4870,8179	545,3776	183,1814	4870,8179
ZDT1	Best	0,6312	0,2776	**0,8175**	0,0559	0,2430	0,1332
	Mean	0,7551	0,2776	0,8601	0,0717	0,0975	0,6825
	Worst	0,8869	0,2776	1,0232	0,0912	0,0000	0,9233
	Standart	0,0695	0,0000	0,0432	0,0100	0,1121	0,2263
	Time	662,2493	141,3644	4053,5786	662,2493	141,3644	4053,5786
ZDT2	Best	0,6657	0,2293	**1,0127**	0,0625	0,0000	**2,457E+06**
	Mean	0,7834	0,2293	1,2470	0,0769	0,0000	1,324E+11
	Worst	0,8453	0,2293	1,3996	0,0883	0,0000	2,621E+12
	Standart	0,0430	0,0000	0,1175	0,0061	0,0000	5,857E+11
	Time	702,9784	157,1703	4786,8025	702,9784	157,1703	4786,8025
ZDT3	Best	0,8497	**0,9671**	0,7542	0,1825	**1,0000**	0,0000
	Mean	0,9385	0,9671	0,9655	0,2211	0,9500	1,5494
	Worst	1,0533	0,9671	1,3790	0,2711	0,0000	7,4420
	Standart	0,0580	0,0000	0,1483	0,0264	0,2236	1,8608
	Time	442,1689	142,1434	4218,0993	442,1689	142,1434	4218,0993
ZDT4	Best	0,6315	0,2770	**0,9092**	0,0535	**0,2690**	0,2266
	Mean	0,6956	0,2770	0,9555	0,0690	0,1332	0,5705
	Worst	0,7693	0,2770	1,0041	0,0886	0,0000	1,0606
	Standart	0,0379	0,0000	0,0304	0,0092	0,1241	0,2767
	Time	347,9827	152,1434	4269,7994	347,9827	152,1434	4269,7994
ZDT6	Best	0,7094	0,1692	**0,8813**	0,0697	0,0000	**3,938E+21**
	Mean	0,8393	0,5665	0,9938	0,0864	0,0000	3,662E+40
	Worst	0,9498	0,8606	1,0613	0,1085	0,0000	4,127E+41
	Standart	0,0714	0,3338	0,0378	0,0113	0,0000	1,034E+41
	Time	683,4443	207,2610	5032,5692	683,4443	207,2610	5032,5692

4. Conclusion

In the study, the performance of current multi-objective optimization algorithms on unconstrained comparison functions and unconstrained engineering problems has been evaluated. While evaluating, MOAHA, which is the most up-to-date multi-objective optimization algorithm, NSGA-II, which is a very useful algorithm with its success in multi-objective problems, and MOGoldSA algorithms, which we brought to the literature in our master's study and which revealed very efficient results, were used. The studies were carried out equally for each algorithm and were evaluated according to the success metrics based on the literature while comparing the performance. According to this evaluation, on unconstrained comparison functions and engineering problems, MOGoldSA according to GD and S criteria, NSGA-II according to MS, SP and HV criteria, and finally MOAHA multi-objective optimization algorithms according to RGD criteria were superior.

References

[1] K. Murty, “Optimization Models For Decision Making”,http://wwwpersonal.umich.edu/~murty/books/opti_model/junior-0.pdf),, 2003.
[2] E. Eröz and E. Tanyildizi, "Çok Amaçlı Metasezgisel Optimizasyon Algoritmalarının Performans Karşılaştırması," 2019 International Artificial Intelligence and Data Processing Symposium (IDAP), 2019, pp. 1-11, doi: 10.1109/IDAP.2019.8875955.
[3] E. Talbi, “Metaheuristic: Design to Implementation, 2nd Edition”, New Jersey: Wiley, 2009.
[4] Alataş. B., “Kaotik Haritalı Parçacık Sürü Optimizasyonu Algoritmaları Geliştirme” 2007.
[5] S. Mirjalili,”Dragonfly Algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems” Neural Comput & Applic 27, pp. 1053-1073, 2016.
[6] S. Mirjalili,”Dragonfly Algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems” Neural Comput & Applic 27, pp. 1053-1073, 2016.
[7] S. M. P. J. S. S., “Multi-objective ant lion optimizer: a multi-objective optimization algorithm for solving engineering problems,” Appl Intell, DOI 10.1007/s10489-016-0825-8, 2016.
[8] K. P. Rainer S., “Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces,” Journal of Global Optimization 11, pp. 341-359, 1997.
[9] S. S. S. M. M. L. S. C. Seyedali M., “Multi-objective grey Wolf optimizer:A novel Algorithm for multi-criterion Optimization,” Expert SystemsWithApplications47, pp. 106-119, 2016.
[10] G. P. M. L. C.A.C. Coello,” Handling multiple objectives with particle swarm Optimization,” IEEE Transactions on Evolutionary Computation Volume: 8, Issue: 3 , 2004.
[11] K. D. N. Sirinivas, “Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms,” Journal of Evolutionary Computation, Vol. 2, No. 3, pp. 221-248, 1994.
[12] Zhao W, Zhang Z, Mirjalili S, Wang L, Khodadadi N, Mirjalili SM, “An effective multi-objective artificial hummingbird algorithm with dynamic elimination-based crowding distance for solving engineering design problems”, Computer Methods in Applied Mechanics and Engineering, 2022; 398, https://doi.org/10.1016/j.cma.2022.115223
[13] A. B., “Kaotik Haritalı Parçacık Sürü Optimizasyonu Algoritmaları Geliştirme, Doktora
[14] Tanyıldızı E, Demir G. Golden Sine Algorithm: A Novel Math-Inspired Algorithm. Adv Electr Comput En, 2017; 17(2):71-78.
[15] W. Zhao, L. Wang, S. Mirjalili, Artifical hummingbird algorithm: A new bia-inspired optimizer with its engineering applications, Comput. Methods Appl. Mech. Engrg. 388 (2022) 318
[16] .K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," in IEEE Transactions on Evolutionary Computation, vol. 6, no. 2, pp. 182-197, April 2002, doi: 10.1109/4235.996017.
[17] M.B. Patil, Using external archive for improved performance in multi-objective optimization, 2018, arXiv preprint arXiv:1811.09196.
[18] Patil, M.B. (2018). Using External Archive for Improved Performance in Multi-Objective Optimization. ArXiv, abs/1811.09196.
[19] Deb K. Multi-objective optimization using evolutionary algorithms. New York: John Wiley&Sons, 2001.
[20] Van Veldhuizen, DA and Lamont GB. Multiobjective evolutionary algorithm research: A history and analysis. Technical Report TR-98-03, Department of Electrical and Computer Engineering, Graduate School of Engineering, Air Force Institute of Technology, WrightPatterson AFB, Ohio, 1998
[21] Zitzler E. Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications, Ph.D Thesis, Swiss Federal Institute of Technology, Switzerland. 1999.
[22] Miettinen K. Nonlinear multiobjective optimization, Kluwer Academic Publishers, Boston: SpringerScience& Bus Media. 1999.
[23] Schott JR. Fault Tolerant Design Using Single and Multi-Criteria Genetic Algorithms. Master of Science Thesis, Massachusetts Institute of Technology, Cambridge, 1995.