İSTANBUL UNIVERSITY CERRAHPAŞA

ORIGINAL ARTICLE
Electrica 2022; 22(2): 143-159

SCANM: A Novel Hybrid Metaheuristic Algorithm and its Comparative Performance Assessment

Murat Kayri[1], Cengiz İpek[2], Davut İzci[3], Erdal Eker[4]

[1]Department of Computer and Instructional Technology Education, Van Yüzüncü Yıl University, Van, Turkey
[2]Department of Statistics, Van Yüzüncü Yıl University, Van, Turkey
[3]Department of Electronics & Automation, Batman University, Batman, Turkey
[4]Vocational School of Social Sciences, Muş Alparslan University, Muş, Turkey

Cite this article as: M. Kayrı, C. İpek, D. İzci and E. Eker, "SCANM: A novel hybrid metaheuristic algorithm and its comparative performance assessment," *Electrica.*, 22(2), 143-159, 2022.

ABSTRACT

This paper proposes a novel sine–cosine and Nelder–Mead (SCANM) algorithm which hybridizes the sine–cosine algorithm (SCA) and Nelder–Mead (NM) local search method. The original version of SCA is prone to early convergence at the local minimum. The purpose of the SCANM algorithm is to overcome this issue. Thus, it aims to overcome this issue with the employment of the NM method. The SCANM algorithm was firstly compared with the SCA algorithm through 23 well-known test functions. The statistical assessment confirmed the better performance of the proposed algorithm. The comparative convergence profiles further demonstrated the significant performance improvement of the proposed SCANM algorithm. Besides, a non-parametric test was performed, and the results that showed the ability of the proposed approach were not by coincidence. A popular and well-performed metaheuristic algorithm known as grey wolf optimization was also used along with the recent and promising two other algorithms (Archimedes optimization and Harris hawks optimization) to comparatively demonstrate the performance of the SCANM algorithm against well-known classical benchmark functions and CEC 2017 test suite. The comparative assessment showed that the SCANM algorithm has promising performance for optimization problems. The non-parametric test further verified the better capability of the proposed SCANM algorithm for optimization problems.
Index Terms—Sine–cosine algorithm, Nelder–Mead simplex search method, hybrid algorithm, benchmark functions

I. INTRODUCTION

The optimization process follows a path that allows a problem to reach a global optimum [1, 2] and such a process has so far been used to solve real-world engineering, finance, and scientific problems which are expressed by mathematical models [3–5]. Optimization techniques can be classified as derivative and non-derivative approaches depending on the derivatives of the objective function which are required to calculate the optimum value. The metaheuristic algorithms are among the non-derivative approaches and have recently been used as the most effective tools for optimization problems. Such algorithms operate based on imitating individuals or swarms of particles with swarm intelligence along with theorems and phenomena based on scientific laws with certain mathematical modeling [3].

In general, metaheuristic approaches initiate the optimization process by generating a random set of candidate solutions and developing them as candidate solutions for a particular problem [6]. Despite the superior performance compared to computational optimization approaches, it is not always feasible to successfully solve all kinds of problems with metaheuristic algorithms. The latter case is well explained by the “no free lunch theorem” [7] which states that one cannot be sure that the success of an algorithm in solving a given set of problems can solve every optimization problem of different nature and type. Such a reality allows researchers to propose new optimization techniques or improve existing algorithms to solve a wider range of problems.

Since metaheuristic approaches have stochastic natures, it is feasible to encounter inefficient solutions for complex problems due to local minimum stagnation. Therefore, a good balance between global and local search stages of metaheuristic algorithms is crucial as such a balance makes them capable of solving different types of problems effectively [8]. To achieve an improved balance, a hybridization technique can be employed [9]. Hybridization has three important

Corresponding author:
Erdal Eker

E-mail:
e.eker@alparslan.edu.tr

Received: September 13, 2021

Accepted: March 22, 2022

DOI: 10.54614/electrica.2022.21112

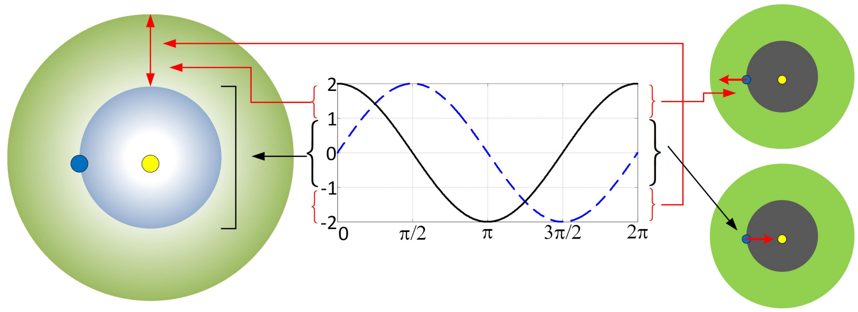

Fig. 1. Conceptual model of the effects of sine and cosine functions in the range [−2, 2].

advantages such as solving large problems that metaheuristic algorithms fail to achieve, faster performance in problem-solving, and providing robust algorithms [10].

The sine–cosine algorithm (SCA) was shown to be an effective metaheuristic approach in solving optimization problems [11]. The SCA has the advantages of convergence speed, powerful neighborhood search capabilities, and a robust global search that allows the algorithm to solve different problems effectively. Nevertheless, certain weaknesses can also be observed in SCA. For example, a random number ranging from 0 to 2 may make some regions challenging for SCA during the search for a better solution around the current solution [12–14]. In that sense, the SCA can be hybridized with Nelder–Mead (NM) local search algorithm as the latter one has the ability to produce a more optimal solution by improving the global solution point in the exploitation stage [15]. So far, there have been different applications adopting hybrid algorithms that use NM simplex search and some of the recent ones can be found in refs [16–22]. Therefore, this study proposes an SCA with NM (SCANM) hybrid algorithm in order to accelerate the convergence to the optimal solution by moving the current solution toward the best solution. In this way, it is feasible to achieve better results compared to similar optimization algorithms.

To assess the performance of the proposed SCANM algorithm, classical unimodal and multimodal benchmark functions with different dimensions [23, 24] were employed. The classical benchmark-related performance assessment has shown that the performance of the SCA is improved significantly by operating the NM method with the specific configuration which confirms the superiority of the proposed SCANM algorithm. Besides, Wilcoxon's signed-rank test further verified the overall superiority of the proposed approach over the original form of SCA. In addition to comparative analysis with the original form of SCA, the proposed SCANM was compared with grey wolf optimization (GWO) [25], Archimedes optimization algorithm (AOA) [26], and Harris hawks optimization (HHO) [27] algorithms in order to further demonstrate the greater performance characteristics of the proposed algorithm against the most popular and recently proposed promising metaheuristic algorithms. The latter comparative assessments were performed against unimodal fixed-dimension and multimodal fixed-dimension along with shifted, rotated, hybrid, and composite benchmark functions from CEC 2017 test suite. The evaluated convergence curves and boxplot figures have confirmed that the proposed SCANM algorithm has better performance characteristics compared to other available and well-performing metaheuristic algorithms. Similar to the assessment carried out against the SCA algorithm, a non-parametric test was also performed against the layer-listed algorithms which further verified the greater performance of the SCANM.

II. SCA, NM SIMPLEX METHOD, AND SCANM HYBRID ALGORITHM

A. Sine–Cosine Algorithm

The optimization algorithm of SCA mimics the mathematical functions of sine and cosine for solving the problems [11]. In SCA, a candidate set of solutions is generated and then updated according to

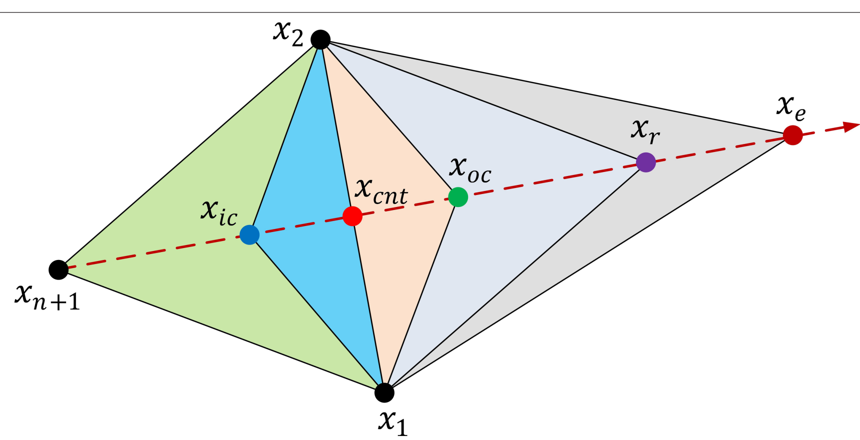

Fig. 2. Visual description of the NM simplex method. NM, Nelder–Mead algorithm.

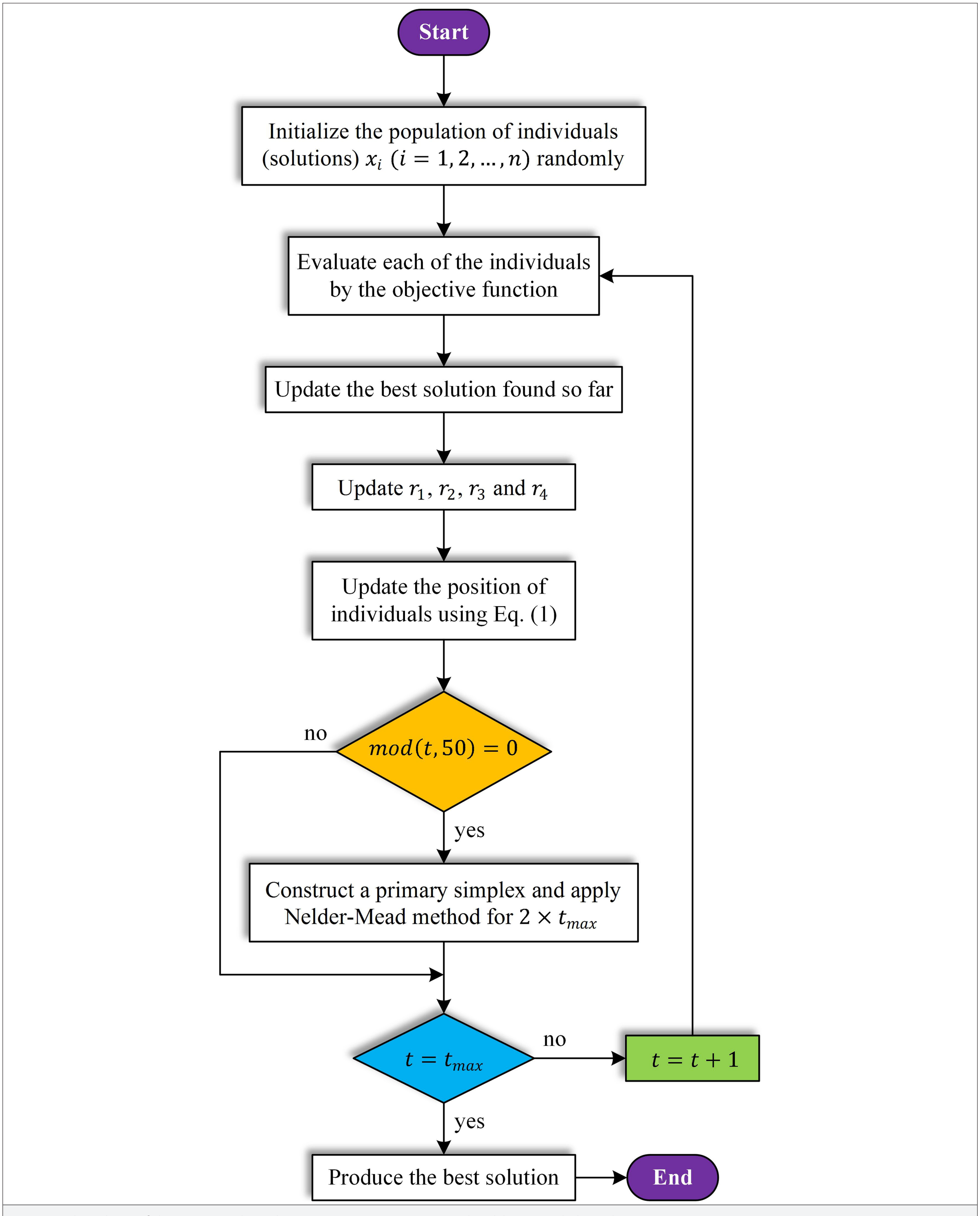

Fig. 3. Flowchart of the SCANM algorithm. SCANM, sine–cosine and Nelder–Mead algorithm.

Q6

TABLE I. ADOPTED CLASSICAL BENCHMARK FUNCTIONS (CF)

Function Number	Name	Range	Dimension (*D*)	Global Minima (f_{min})
CF1	Sphere	[−100, 100]	10	0
CF2	Schwefel 2.22	[−10, 10]	10	0
CF3	Schwefel 1.2	[−100, 100]	10	0
CF4	Schwefel 2.21	[−100, 100]	10	0
CF5	Rosenbrock	[−30, 30]	10	0
CF6	Step	[−100, 100]	10	0
CF7	Quartic	[−1.28, 1.28]	10	0
CF8	Schwefel	[−500, 500]	10	−418.9829**D*
CF9	Rastrigin	[−5.12, 5.12]	10	0
CF10	Ackley	[−32, 32]	10	0
CF11	Griewank	[−600, 600]	10	0
CF12	Penalized	[−50, 50]	10	0
CF13	Penalized 2	[−50, 50]	10	0
CF14	Foxholes	[−65.536, 65.536]	2	0.998004
CF15	Kowalik	[−5, 5]	4	0.0003075
CF16	Six-hump	[−5, 5]	2	−1.0316285
CF17	Camel back branin	[−5, 0], [10, 15]	2	0.398
CF18	Goldstein-price	[−2, 2]	2	3
CF19	Hartman 3	[0, 1]	3	−3.862782
CF20	Hartman 6	[0, 1]	6	-3.32236
CF21	Shekel5	[0, 10]	4	−10.1532
CF22	Shekel7	[0, 10]	4	−10.4029
CF23	Shekel10	[0, 10]	4	−10.5364

the sine and cosine functions by fluctuating outward or toward the target to create a new swarm as shown in Eq. (1):

$$x_i^{t+1} = \begin{cases} x_i^t = r_1 * sin(r_2) * \left| r_3 p_i^t - x_i^t \right|, r_4 < 0.5 \\ x_i^t = r_1 * cos(r_2) * \left| r_3 p_i^t - x_i^t \right|, r_4 \geq 0.5 \end{cases} \quad (1)$$

where *t* represents the current iteration and *i* stands for the dimension at *t*. In the respective equation, P stands for the destination direction. The parameters r_1, r_2, r_3 and r_4 determine the direction of movement of the next position which may be toward inside or outside of the area between the solution and the target. Fig. 1 illustrates the conceptual model that demonstrates the effects of the tine and cosine functions.

B. Nelder–Mead Simplex Method

This is a direct search method developed by Nelder and Mead [15] in order to solve minimization problems. In this algorithm, vertices of x_1, x_2, …, x_{n+1} are generated and the respective fitness function values of $fit(x_1)$, $fit(x_2)$, …, $fit(x_{n+1})$ are sorted in ascending order. In this way, the optimal vertex of x_1 is identified. Four different operations (reflection, expansion, contraction, and shrinkage) are used to replace the worst vertex (x_{n+1}). The reflection point of x_r is identified as in Eq. (2):

$$x_r = x_{cnt} + \rho\left(x_{cnt} - x_{n+1}\right) \quad (2)$$

where ρ and x_{cnt} are, respectively, the reflection coefficient and the centroid (excluding the x_{n+1} vertex). The x_r is then expanded across the search space as defined in Eq. (3):

$$x_e = x_{cnt} + \gamma\left(x_r - x_{cnt}\right) \quad (3)$$

where γ is the expansion coefficient and x_e is the expansion point. For x_c (contraction) and x_s (shrinkage), the following equations are respectively used:

$$x_c = x_{cnt} + \beta\left(x_{n+1} - x_{cnt}\right) \quad (4)$$

$$x_s = x_1 + \delta\left(x_i - x_1\right), i = 2, 3, \ldots, n+1 \quad (5)$$

TABLE II. COMPARISON OF SCA AND SCANM AGAINST UNIMODAL CLASSICAL FUNCTIONS

Function Number	Metrics	SCA	SCANM
CF1	Mean	2.2557E−11	8.2358E−28
	Best	2.3119E−17	**5.1386E−40**
	Worst	5.7183E−10	**2.4690E−26**
	Std. Dev.	1.0411E−10	**4.5076E−27**
CF2	Mean	2.1463E−09	**8.5287E−15**
	Best	3.0693E−14	**5.5911E−19**
	Worst	2.1850E−08	**1.0971E−13**
	Std. Dev.	4.5415E−09	**2.2759E−14**
CF3	Mean	0.0065	**3.0360E−20**
	Best	1.3595E−06	**2.0467E−30**
	Worst	0.0779	**4.5153E−19**
	Std. Dev.	0.0174	**1.0012E−19**
CF4	Mean	**4.4861E−04**	9.1492E−03
	Best	1.7932E−06	**1.1349E−06**
	Worst	**0.0019**	0.0681
	Std. Dev.	**4.9629E−04**	0.014
CF5	Mean	7.39858	**1.3288**
	Best	6.3123	**0**
	Worst	8.1216	**3.9866**
	Std. Dev.	**0.421**	1.9114
CF6	Mean	0.4341	**3.7430E−31**
	Best	0.1378	**3.0815E−32**
	Worst	0.8618	**1.1925E−30**
	Std. Dev.	0.1488	**2.5374E−31**
CF7	Mean	**0.0038**	0.5015
	Best	**3.7334E−05**	0.0567
	Worst	**0.0123**	0.9106
	Std. Dev.	**0.0031**	0.2937

SCA, sine–cosine algorithm; SCANM, sine–cosine and Nelder–Mead algorithm.

Here, δ is the shrinkage coefficient whereas β is the contraction. Fig. 2 illustrates the operation of NM from a geometrical point of view.

C. Proposed SCANM Hybrid Algorithm

As mentioned earlier, SCA may end up with poor convergence for complex optimization tasks which causes unsatisfactory results. Considering this fact, the proposed algorithm aims to increase the exploitation capability so that it can perform satisfactorily for complex optimization tasks. In this regard, the hybrid SCANM algorithm proposed by this study uses SCA to explore the search space whereas employs NM to carry out the local search. In the proposed SCANM algorithm, the search process starts with the SCA, and this is followed by passing the best solution of the current iteration to the NM method in order to improve the solution quality. After every 50 iterations of the SCA, the NM method operates twice as much as the number of SCA meaning that for 500 iterations of SCA, NM processes 1000 iterations. The detailed flowchart of the proposed SCANM algorithm is shown in Fig. 3.

III. EXPERIMENTAL RESULTS

This section provides the details of the employed classical benchmark functions and the test functions chosen from CEC 2017 benchmark suite. The performance of the proposed hybrid SCANM algorithm

TABLE III. COMPARISON OF SCA AND SCANM AGAINST MULTIMODAL CLASSICAL FUNCTIONS

Function number	Metrics	SCA	SCANM
CF8	Mean	−2.14E+03	**−2.50E+03**
	Best value	−2.55E+03	**-2.94E+03**
	Worst value	−1.83E+03	**−2.01E+03**
	Std. Dev.	**152.6469**	281.4528
CF9	Mean	**2.5482**	7.3295
	Best value	**0**	**0**
	Worst value	**14.4277**	32.8334
	Std. Dev.	**4.4909**	9.3096
CF10	Mean	6.8069E−05	**3.0040E−12**
	Best value	9.2668E−10	**7.9936E−15**
	Worst value	9.3053E−04	**6.7765E−11**
	Std. Dev.	2.2722E−04	**1.2289E−11**
CF11	Mean	**0.0628**	0.2921
	Best value	4.7629E−14	**0**
	Worst value	**0.754**	0.8101
	Std. Dev.	**0.1556**	0.2575
CF12	Mean	**0.0959**	0.8583
	Best value	0.0306	**4.7116E−32**
	Worst value	**0.1858**	17.9486
	Std. Dev.	**0.0457**	3.5277
CF13	Mean	0.3276	**0.1007**
	Best value	0.1836	**1.0224E−31**
	Worst value	0.513	**0.4364**
	Std. Dev.	**0.0858**	0.0925

SCA, sine–cosine algorithm; SCANM, sine–cosine and Nelder–Mead algorithm.

TABLE IV. COMPARISON OF SCA AND SCANM AGAINST SMALL SIZE CLASSICAL FUNCTIONS

Function number	Metrics	SCA	SCANM
CF14	Mean	**1.5277**	1.5781
	Best value	0.998	**0.998**
	Worst value	2.9821	**2.9821**
	Std. Dev.	0.892	**0.8822**
CF15	Mean	0.001	**3.7264E−04**
	Best value	3.6324E−04	**3.0749E−04**
	Worst value	**0.0016**	0.0023
	Std. Dev.	**3.5609E−04**	3.5687E−04
CF16	Mean	−1.0316	**−1.0316**
	Best value	−1.0316	**−1.0316**
	Worst value	**−1.0314**	−1.0316
	Std. Dev.	4.3258E−05	**1.8278E−15**
CF17	Mean	0.4008	**0.3979**
	Best value	0.3979	0.3979
	Worst value	0.4227	**0.3979**
	Std. Dev.	0.0047	**6.1487E−16**
CF18	Mean	3.0001	**3**
	Best value	3	**3**
	Worst value	3.0005	**3**
	Std. Dev.	1.1150E−04	**9.9827E−15**
CF19	Mean	−3.8534	**−3.8628**
	Best value	−.8604	**−3.8628**
	Worst value	−3.8483	**−3.8628**
	Std. Dev.	0.0022	**1.3550E−15**
CF20	Mean	−2.8724	**−2.8724**
	Best value	−3.2075	**−3.322**
	Worst value	−1.5709	**−3.2031**
	Std.Dev.	0.377	**0.057**
CF21	Mean	−2.1905	**−5.7654**
	Best value	−4.8951	**−10.1532**
	Worst value	−0.4965	**−2.6305**
	Std.Dev.	**1.7599**	2.3787
CF22	Mean	−3.4178	**−5.6041**
	Best value	−5.4807	**−10.4029**
	Worst value	−0.9054	**−2.7519**
	Std. Dev.	**1.5206**	2.5648
CF23	Mean	−3.5113	**−7.0768**
	Best value	−6.7781	**−10.5364**
	Worst value	−0.5557	**−2.4217**
	Std. Dev.	**1.8857**	2.9327

SCA, sine–cosine algorithm; SCANM, sine–cosine and Nelder–Mead algorithm.

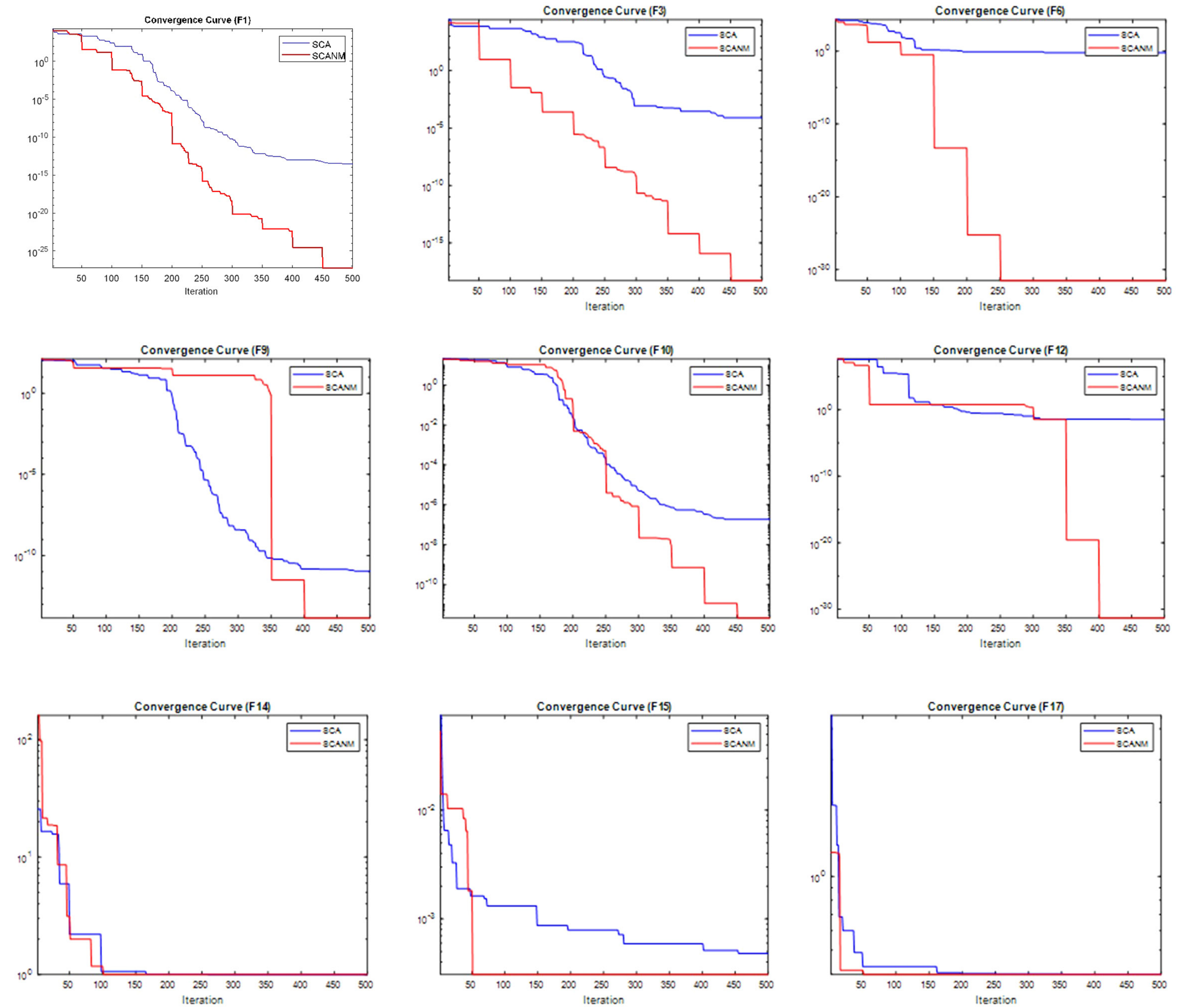

Fig. 4. Comparative convergence curves of SCA and SCANM algorithms obtained from unimodal (top row), multimodal (middle row), and multimodal small size (bottom row) classical benchmark functions. SCA, sine–cosine algorithm; SCANM, sine–cosine and Nelder–Mead algorithm.

was evaluated comparatively against those benchmark functions in terms of statistical measures. In this way, a better balance between the explorative and exploitative phases of the proposed SCANM algorithm is demonstrated. Besides, the Wilcoxon signed-rank nonparametric test is also discussed in this section. For the purpose of the visual observation, the convergence curves and the boxplot figures are also provided and discussed. All algorithms adopted in this study were run 30 times with 500 iterations and population size of 30 for all functions using the MATLAB 2016 software package.

A. The Performance of SCANM Against SCA Using Classical Benchmark Functions

In this study, the classical benchmark functions listed in Table I were adopted in order to observe the performance of the proposed hybrid SCANM algorithm against the original form of SCA. The benchmark functions of CF1–CF7 are unimodal functions that have only one global solution and no local solution. Thus, they are used to examine the algorithms in terms of convergence rate. On the other hand, the benchmark functions from CF8 to CF13 are multimodal functions that have local minima. These functions can be used to assess the algorithm's ability in terms of escaping the weak local optimum and achieving a near-global optimum. Finally, the benchmark functions of CF14–CF23 are of small sizes which also have several local minimums. The latter appears to be relatively simple compared to multimodal functions (CF8–CF13).

The benchmark functions given in Table II were solved by the original SCA and the proposed SCANM algorithms. Then the results were compared in terms of mean, best, worst, and standard deviation values. The obtained numerical results are listed in Tables II, III, and IV for unimodal, multimodal, and small size classical benchmark functions. Considering the numerical results provided in the respective tables, apart from CF7 unimodal and CF9 multimodal benchmark functions, the proposed SCANM algorithm performs better.

Besides the numerical values, the comparative convergence profiles of some benchmark functions from unimodal, multimodal, and

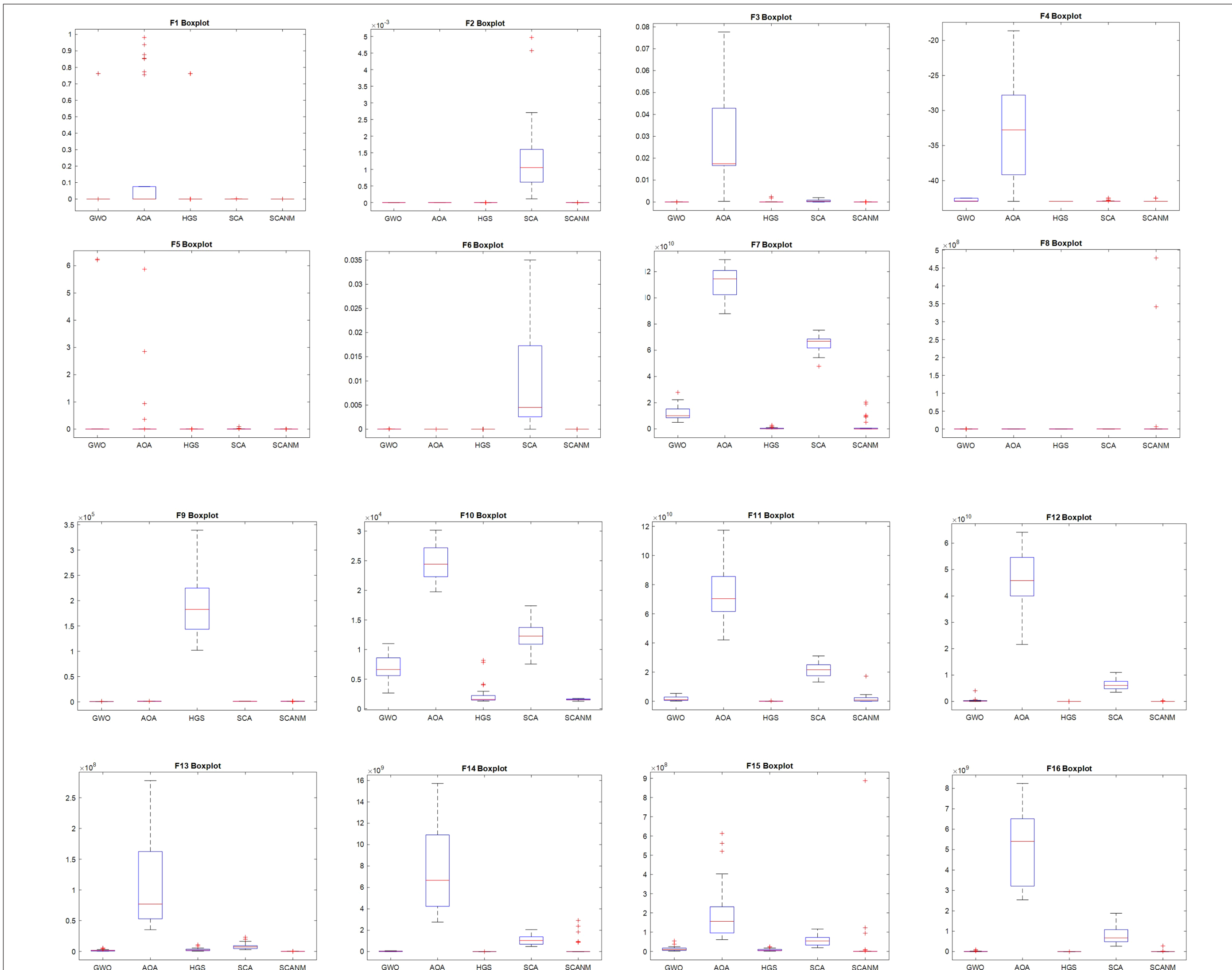

Fig. 5. Boxplots for F1–F17 are defined in Table IV.

multimodal small size types are also illustrated in Figs. 4, 5, and 6. The proposed SCANM can be observed to be better overall in terms of overcoming the stagnation in the local minima and reaching the global optimum point step by step. Moreover, Table V provides the Wilcoxon signed-rank results where $\square$, $\bigtriangledown$, and $\approx$ represents win, lost, and tie. The Wilcoxon test provides a good opportunity for performance assessments since it is a non-parametric statistical hypothesis test that can be used when the results are not assumed to be normally distributed. The paired Wilcoxon signed-rank test at $\alpha = 0.05$ was adopted to compare the significance of the two algorithms in this study. Except for CF4, CF7, CF9, and CF11, the superiority of the proposed SCANM can clearly be observed.

B. The Performance of SCANM Against Other Algorithms Using Different Benchmark Functions

In the previous section, the improved performance of the proposed SCANM has comparatively been demonstrated against the original form of SCA using classical benchmark functions. In this subsection, the performance of the proposed SCANM algorithm is comparatively demonstrated using other available well-performing and popular GWO along with recently proposed and promising AOA and HHO algorithms using challenging benchmark functions from CEC 2017 test suite [23]. In addition, the original form of the SCA has also been used for comparisons. The reason for employing those algorithms, apart from SCA, can briefly be stated as follows. The GWO is one of the most cited and well-performing metaheuristic optimizers while the AOA and the HHO are the recently developed metaheuristic algorithms for optimization problems and have been demonstrated to be highly successful for various problems [26–28].

The adopted benchmark functions are listed in Table VI which includes unimodal fixed size (F1–F3), multimodal fixed size (F4–F6) along with shifted, rotated, hybrid, and composite (F7–F17) benchmark functions. The listed test functions were used to assess the exploration and exploitation capabilities along with the local minima avoiding the ability of the SCANM algorithm comparatively.

The comparative numerical results for the abovementioned benchmark functions are provided in Tables VII and VIII. The respective

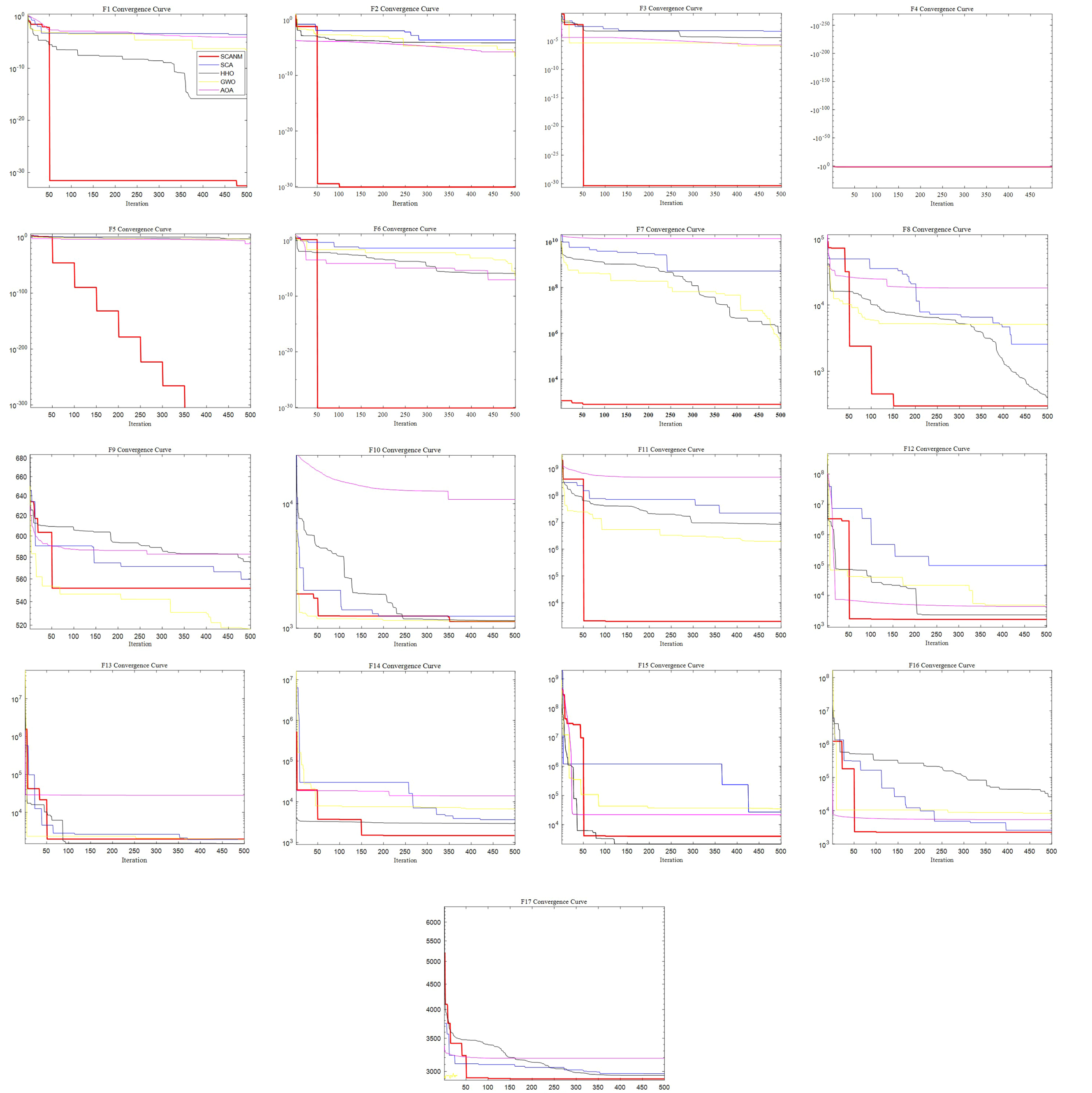

Fig. 6. Convergence curves of F1–F17 functions are defined in Table IV.

tables clearly demonstrate the overall better ability of the proposed SCANM algorithm over other algorithms of GWO, AOA, HHO, and SCA.

In addition, the performance of the proposed SCANM has also comparatively been evaluated against all tests in terms of convergence and boxplot analyses. The boxplot analysis for all test functions is provided in Fig. 5 and the convergence curves are given in Fig. 6. The SCANM algorithm achieves all statistical metrics nearest to each other and around the optimum value. However, some values fall far from the range only for F8. As can be observed in the convergence curves, provided in Fig. 6, the SCANM algorithm converges better than other algorithms despite the difference in function types.

The effectiveness of the SCANM algorithm against other available competitive algorithms has also been measured using Wilcoxon signed-rank test as listed in Table IX. Apart from functions of F9, F10, F14, and F17, SCANM is the winner compared to the GWO algorithm. In terms of AOA, the proposed SCANM is the winner for all functions. Similarly, the proposed SCANM is the winner for test functions (except the functions F10, F11, F12, and F14) compared to the HHO algorithm. Lastly, the proposed SCANM is the winner

TABLE V. WILCOXON SIGNED-RANK TEST RESULTS FOR SCA AND SCANM AGAINST TEST FUNCTIONS OF CF1–CF23

Function	*P*	Results
CF1	1.7344E−06	□
CF2	1.7344E−06	□
CF3	1.7344E−06	□
CF4	1.3595E−04	▽
CF5	1.7344E−06	□
CF6	1.7344E−06	□
CF7	1.7344E−06	▽
CF8	1.9729E−05	□
CF9	0.035	▽
CF10	1.7344E−06	□
CF11	4.4493E−05	▽
CF12	3.5888E−04	□
CF13	2.3534E−06	□
CF14	0.102	≈
CF15	3.1123E−05	□
CF16	1.7344E−06	□
CF17	1.7344E−06	□
CF18	1.7344E−06	□
CF19	1.7344E−06	□
CF20	1.9209E−06	□
CF21	1.7344E−06	□
CF22	3.5888E−04	□
CF23	2.8434E−05	□

TABLE VI. UNIMODAL FIXED-DIMENSION, MULTIMODAL FIXED-DIMENSION, AND CEC 2017 FUNCTIONS

Type	Function	Name	Range	Dim (*D*)	Global Min. ($f_{min.}$)
Unimodal fixed dimension	F1	Beale	[−4.5, 4.5]	2	0
	F2	Booth	[−10, 10]	2	0
	F3	Leon	[−10, 10]	2	0
Multimodal fixed dimension	F4	Chichinadze	[−30, 30]	2	−43.3159
	F5	Helical Valley	[−10, 10]	3	0
	F6	Himmelblau	[−5, 5]	2	0
Benchmark functions from CEC 2017 test Suite	F7	Shifted and Rotated Bent Cigar Function	[−100, 100]	10	100
	F8	Shifted and Rotated Rosenbrock's Function	[−100, 100]	10	300
	F9	Shifted and Rotated Expanded Schaffer's F6 Function	[−100, 100]	10	500
	F10	Hybrid Function 2 (N = 3)	[−100, 100]	10	1100
	F11	Hybrid Function 3 (N = 3)	[−100, 100]	10	1200
	F12	Hybrid Function 4 (N = 4)	[−100, 100]	10	1300
	F13	Hybrid Function 5 (N = 4)	[−100, 100]	10	1400
	F14	Hybrid Function 6 (N = 4)	[−100, 100]	10	1500
	F15	Hybrid Function 6 (N = 5)	[−100, 100]	10	1800
	F16	Hybrid Function 6 (N = 6)	[−100, 100]	10	1900
	F17	Composite Function 5 (N = 5)	[−100, 100]	10	2500

TABLE VII. PERFORMANCES OF ALGORITHMS THROUGH BENCHMARKS OF UNIMODAL AND MULTIMODAL FIXED FUNCTIONS

Function	Metrics	GWO	AOA	HHO	SCA	SCANM
F1	Mean	5.0805E−02	3.6083E−01	1.5458E−10	3.7012E−04	**2.0543E−33**
	Best value	1.1117E−09	6.3149E−05	0	4.0369E−05	**0**
	Worst value	7.6207E−01	1.4053E+00	2.4889E−09	1.5709E−03	**6.1630E−32**
	Std. Dev.	1.9334E−01	4.6680E−01	5.3855E−10	4.0199E−04	**1.1252E−32**
F2	Mean	6.1768E−07	7.0503E−07	5.3274E−05	1.5779E−03	**5.2591E−32**
	Best value	6.2710E−09	6.2074E−09	1.3485E−09	1.8847E−05	**0**
	Worst value	2.4683E−06	1.7325E−06	3.3179E−04	4.7345E−03	**7.8886E−31**
	Std. Dev.	5.8463E−07	5.1931E−07	7.4808E−05	1.1903E−03	**2.0014E−31**
F3	Mean	2.4468 E−06	2.7302E−02	2.8463E−05	6.9055E−04	**6.5738E−33**
	Best value	5.6232E−08	1.9811E−06	4.4373E−31	5.8615E−06	**0**
	Worst value	1.2893E−05	8.5024E−02	1.4730E−04	2.9711E−03	**4.9304E−32**
	Std. Dev.	3.1629E−06	2.6490E−02	4.2952E−05	6.5909E−04	**1.7047E−32**
F4	Mean	−4.2857E+01	−3.5285E+01	−4.2810E+01	−4.2895E+01	**−4.2900E+01**
	Best value	−4.2944E+01	−4.2944E+01	−4.2944E+01	−4.2944E+01	**−4.2944E+01**
	Worst value	−4.2497E+01	−1.9097E+01	−4.2497E+01	−4.2497E+01	**−4.2497E+01**
	Std. Dev.	1.6556 E−01	7.5345E+00	2.0808E−01	**9.8564E−02**	1.3646E−01
F5	Mean	9.7870 E−04	2.6352E−01	2.0524E−02	5.0601E−03	**3.9525E−323**
	Best value	2.4530E−04	6.7526E−13	4.4505E−07	5.8203E−06	**0**
	Worst value	2.8038E−03	2.2867E+00	2.0533E−01	2.6192E−02	**8.2015E−322**
	Std. Dev.	6.4244E−04	6.8319E−01	4.0824E−02	6.4953E−03	**0**
F6	Mean	9.7922E−06	9.0476E−07	5.6097E−06	1.7989E−02	**4.9961E−31**
	Best value	2.9040E−07	1.3020E−08	2.3886E−16	8.7665E−05	**0**
	Worst value	4.7388E−05	3.7600E−06	7.1324E−05	7.2318E−02	**3.1554E−30**
	Std. Dev.	1.1091E−05	9.6477E−07	1.5322E−05	1.8820E−02	**8.1517E−31**

SCA, sine–cosine algorithm; SCANM, sine–cosine and Nelder–Mead algorithm; GWO, grey wolf optimization; AOA, Archimedes optimization algorithm; HHO, Harris hawks optimization.

TABLE VIII. PERFORMANCES OF ALGORITHMS THROUGH BENCHMARK FUNCTION FROM CEC 2017

Function	Metrics	GWO	AOA	HHO	SCA	SCANM
F7	Mean	2.7577E+07	8.9130E+09	**1.0599E+06**	8.9800E+08	7.5594E+02
	Best value	5.3703E+04	3.2051E+09	3.3433E+05	5.1802E+08	**7.2656E+02**
	Worst value	3.3377E+08	1.7995E+10	**4.6279E+06**	1.3584E+09	7.6947E+02
	Std. Dev.	8.4233E+07	4.4830E+09	**8.3496E+05**	2.5295E+08	9.6575E+00
F8	Mean	3.1522E+03	1.2071E+04	7.2816E+02	2.9002E+03	**3.0000E+02**
	Best value	3.1844E+02	4.6399E+03	3.2810E+02	9.5592E+02	**300**
	Worst value	1.5581E+04	1.7988E+04	2.1162E+03	8.2647E+03	**3.0000E+02**
	Std. Dev.	3.3985E+03	2.9694E+03	3.9479E+02	1.6637E+03	**2.5985E–13**
F9	Mean	**5.2215E+02**	5.6657E+02	5.5299E+02	5.5611E+02	5.3684E+02
	Best value	**5.0603E+02**	5.2237E+02	5.2049E+02	5.3968E+02	5.1293E+02
	Worst value	**5.4323E+02**	5.9811E+02	5.8622E+02	5.7094E+02	5.6074E+02
	Std. Dev.	9.6594E+00	1.7994E+01	1.9160E+01	**7.4380E+00**	1.1368E+01
F10	Mean	**1.1432E+03**	4.7633E+03	1.2022E+03	1.2542E+03	1.1918E+03
	Best value	1.1101E+03	1.3422E+03	1.1236E+03	1.1487E+03	**1.1050E+03**
	Worst value	**1.2217E+03**	1.1582E+04	1.3683E+03	1.4067E+03	1.7438E+03
	Std. Dev.	**2.9470E+01**	3.3163E+03	6.5688E+01	6.0652E+01	1.1493E+02
F11	Mean	5.8932E+05	2.6379E+08	3.8260E+06	2.9540E+07	**3.8415E+04**
	Best value	1.4111E+04	2.3039E+04	3.3085E+04	3.7971E+06	**1.4825E+03**
	Worst value	3.1952E+06	1.7369E+09	1.8681E+07	1.1844E+08	**9.6781E+05**
	Std. Dev.	7.9914E+05	3.7306E+08	4.3306E+06	2.9334E+07	**1.7646E+05**
F12	Mean	1.2097E+04	1.2045E+04	1.5438E+04	6.4259E+04	**3.2664E+03**
	Best value	3.4951E+03	3.6035E+03	2.2459E+03	3.2823E+03	**1.3334E+03**
	Worst value	3.1842E+04	3.0544E+04	4.7288E+04	3.8177E+05	**1.5539E+04**
	Std. Dev.	7.3391E+03	8.6250E+03	1.2496E+04	7.1685E+04	**4.2293E+03**
F13	Mean	3.5630E+03	8.2997E+03	1.8110E+03	2.2445E+03	**1.6090E+03**
	Best value	1.4739E+03	1.4630E+03	1.4911E+03	1.5060E+03	**1.4433E+03**
	Worst value	7.2449E+03	2.8196E+04	3.0128E+03	5.1392E+03	**1.9760E+03**
	Std. Dev.	2.0684E+03	8.8217E+03	3.7711E+02	8.9221E+02	**1.4013E+02**
F14	Mean	7.9933E+03	1.8543E+04	8.6255E+03	**4.0519E+03**	8.3746E+03
	Best value	1.5836E+03	4.1909E+03	4.2116E+03	1.9491E+03	**1.5364E+03**
	Worst value	2.7401E+04	4.4210E+04	**1.3419E+04**	8.4934E+03	4.1752E+04
	Std. Dev.	6.9526E+03	8.5143E+03	2.6742E+03	**1.8663E+03**	1.3211E+04
F15	Mean	2.8458E+04	7.2500E+06	1.8167E+04	4.4106E+05	**2.7643E+03**
	Best value	2.8285E+03	2.9501E+03	2.1558E+03	2.6956E+04	**1.8295E+03**
	Worst value	5.3095E+04	1.9279E+08	4.0875E+04	1.6681E+06	**8.4551E+03**
	Std. Dev.	1.4356E+04	3.5315E+07	1.3349E+04	4.5058E+05	**1.4828E+03**

TABLE VIII. PERFORMANCES OF ALGORITHMS THROUGH BENCHMARK FUNCTION FROM CEC 2017 (*CONTINUED*)

Function	Metrics	GWO	AOA	HHO	SCA	SCANM
F16	Mean	1.0214E+04	1.1504E+05	2.0217E+04	9.5786E+03	**2.6841E+03**
	Best value	1.9498E+03	1.9369E+03	1.9901E+03	1.9945E+03	**1.9126E+03**
	Worst value	2.4214E+04	2.3583E+05	1.4068E+05	3.1633E+04	**1.0775E+04**
	Std. Dev.	7.5209E+03	8.1856E+04	2.9057E+04	8.5338E+03	**1.7784E+03**
F17	Mean	2.9378E+03	3.3210E+03	2.9387E+03	3.0101E+03	**2.9359E+03**
	Best value	2.9002E+03	3.0707E+03	2.8997E+03	2.9569E+03	**2.8978E+03**
	Worst value	2.9767E+03	4.1075E+03	2.9769E+03	3.9614E+03	3.0394E+03
	Std. Dev.	1.5782E+01	2.3542E+02	1.9934E+01	1.8033E+02	3.5725E+01

SCA, sine–cosine algorithm; SCANM, sine–cosine and Nelder–Mead algorithm; GWO, grey wolf optimization; AOA, Archimedes optimization algorithm; HHO, Harris hawks optimization.

TABLE IX. WILCOXON SIGNED-RANK TEST RESULTS BETWEEN SCANM AND OTHER ALGORITHMS

Function	Metrics	GWO	AOA	HHO	SCA
F1	*P*	1.7344E−06	1.7344E−06	3.7896E−06	1.7344E−06
	Result	□	□	□	□
F2	*P*	1.7344E−06	1.7344E−06	1.7344E−06	1.7344E−06
	Result	□	□	□	□
F3	*P*	1.7344E−06	1.7344E−06	1.7344E−06	1.7344E−06
	Result	□	□	□	□
F4	*P*	1.1138E−03	1.1265E−05	1.3317E−04	2.1053E−03
	Result	□	□	□	□
F5	*P*	1.7344E−06	1.7344E−06	1.7344E−06	1.7344E−06
	Result	□	□	□	□
F6	*P*	1.7344E−06	1.7344E−06	1.7344E−06	1.7344E−06
	Result	□	□	□	□
F7	*P*	1.4839E−03	1.7344E−06	2.7653E−03	1.7344E−06
	Result	□	□	□	□
F8	*P*	1.7344E−06	1.7344E−06	1.7344E−06	1.7344E−06
	Result	□	□	□	□
F9	*P*	2.2248E−04	3.5152E−06	4.8969E−04	4.2857E−06
	Result	▽	□	□	□
F10	*P*	3.6094E−03	1.7344E−06	2.2888E−01	1.1138E−03
	Result	▽	□	▽	□
F11	*P*	3.5888E−04	3.1123E−05	7.9710E−01	3.1123E−05
	Result	□	□	▽	□
F12	*P*	3.4053E−05	9.7110E−05	3.4053E−05	2.6033E−06
	Result	□	□	□	□
F13	*P*	2.0515E−04	2.3534E−06	6.0350E−03	2.1630E−05
	Result	□	□	□	□
F14	*P*	1.4704E−01	2.9575E−03	3.8203E−01	3.9333E−01
	Result	▽	□	▽	▽
F15	*P*	2.1266E−06	1.7344E−06	2.8786E−06	1.7344E−06
	Result	□	□	□	□
F16	*P*	1.4773E−04	2.1266E−06	8.4661E−06	7.6909E−06
	Result	□	□	□	□
F17	*P*	0.7499	1.7344E−06	7.6552E−01	2.1630E−05
	Result	▽	□	▽	□
Rank (□/▽/≈)		13/4/0	17/0/0	13/4/0	16/1/0

SCA, sine–cosine algorithm; SCANM, sine–cosine and Nelder–Mead algorithm; GWO, grey wolf optimization; AOA, Archimedes optimization algorithm; HHO, Harris hawks optimization.

for all test functions, except F14, compared to SCA. Those results confirm the overall greater capability of the proposed SCANM algorithm.

IV. CONCLUSION

In this study, a new hybrid metaheuristic optimization algorithm named SCANM has been proposed for solving optimization problems with different natures. The proposed SCANM algorithm has been used to solve classical, unimodal, multimodal, fixed dimension, shifted, rotated, hybrid, and composite benchmark functions. The obtained statistical results demonstrated the highly competitive structure of the proposed SCANM algorithm. The effectiveness of the SCANM has further been shown by performing Wilcoxon signed-rank test as a non-parametric test. The obtained results have shown that the proposed SCANM hybrid algorithm can be used to solve real-world optimization problems in potential future works.

Peer-review: Externally peer-reviewed.

Author Contributions: Concept – M.K., C.İ., D.İ., E.E.; Design – M.K., C.İ., D.İ., E.E.; Materials – M.K., C.İ., D.İ., E.E.; Data Collection and/or Processing – M.K., C.İ., D.İ., E.E.; Analysis and/or Interpretation – M.K., C.İ., D.İ., E.E.; Literature Review – M.K., C.İ., D.İ., E.E.; Writing – M.K., C.İ., D.İ., E.E.

Declaration of Interests: The authors declare that they have no competing interest.

Funding: The authors declared that this study has received no financial support.

REFERENCES

1. D. Izci, "An Enhanced Slime Mould Algorithm for Function optimization," in 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), 2021, pp. 1–5. [CrossRef]
2. A. Dündar, D. Izci, S. Ekinci, and E. Eker, "A Novel Modified Lévy Flight Distribution Algorithm based on Nelder-Mead Method for Function Optimization," *DÜMF Mühendislik Derg.*, vol. 12, no. 3, pp. 487–496, 2021. [CrossRef]
3. E. Eker, M. Kayri, S. Ekinci, and D. Izci, "A new fusion of ASO with SA algorithm and its applications to MLP training and DC motor speed control," *Arab. J. Sci. Eng.*, vol. 46, no. 4, pp. 3889–3911, 2021. [CrossRef]
4. R. M. Rizk-Allah, "Hybridizing sine cosine algorithm with multi-orthogonal search strategy for engineering design problems," *J. Comput. Des. Eng.*, vol. 5, no. 2, pp. 249–273, 2018. [CrossRef]
5. D. Izci, S. Ekinci, M. Kayri, and E. Eker, "A novel improved arithmetic optimization algorithm for optimal design of PID controlled and Bode's ideal transfer function based automobile cruise control system," *Evol. Syst.*, 2021. [CrossRef]
6. H. Nenavath, D. R. K. Jatoth, and D. S. Das, "A synergy of the sine-cosine algorithm and particle swarm optimizer for improved global optimization and object tracking," *Swarm Evol. Comput.*, vol. 43, pp. 1–30, 2018. [CrossRef]
7. D. H. Wolpert, and W. G. Macready, "No free lunch theorems for optimization," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, 1997. [CrossRef]
8. D. Izci, S. Ekinci, E. Eker, and M. Kayri, "Improved manta ray foraging optimization using opposition-based learning for optimization problems," in International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), 2020, pp. 1–6. [CrossRef]
9. H. Nenavath, and R. K. Jatoth, "Hybridizing sine cosine algorithm with differential evolution for global optimization and object tracking," *Appl. Soft Comput.*, vol. 62, pp. 1019–1043, 2018. [CrossRef]
10. H. N. Fakhouri, A. Hudaib, and A. Sleit, "Hybrid particle swarm optimization with sine cosine algorithm and Nelder–mead simplex for solving engineering design problems," *Arab. J. Sci. Eng.*, vol. 45, no. 4, pp. 3091–3109, 2020. [CrossRef]
11. S. Mirjalili, "SCA: A Sine cosine Algorithm for solving optimization problems," *Knowl. Based Syst.*, vol. 96, pp. 120–133, 2016. [CrossRef]
12. J. Zhang, Y. Zhou, and Q. Luo, "An improved sine cosine water wave optimization algorithm for global optimization," *J. Intell. Fuzzy Syst.*, vol. 34, no. 4, pp. 2129–2141, 2018. [CrossRef]
13. M. Abdel-Basset, R. Mohamed, M. Abouhawwash, R. K. Chakrabortty, and M. J. Ryan, "EA-MSCA: An effective energy-aware multi-objective modified sine-cosine algorithm for real-time task scheduling in multiprocessor systems: Methods and analysis," *Expert Syst. Appl.*, vol. 173, p. 114699, 2021. [CrossRef]
14. S. Ekinci, "Optimal design of power system stabilizer using sine cosine algorithm," *J. Fac. Eng. Archit. Gazi Univ.*, vol. 34, no. 3, pp. 1330–1350, 2019. [CrossRef]
15. J. A. Nelder, and R. Mead, "A simplex method for function minimization," *Comput. J.*, vol. 7, no. 4, pp. 308–313, 1965. [CrossRef]
16. D. Izci, "Design and application of an optimally tuned PID controller for DC motor speed regulation via a novel hybrid Lévy flight distribution and Nelder–Mead algorithm," *Trans. Inst. Meas. Control*, vol. 43, no. 14, pp. 3195–3211, 2021. [CrossRef]
17. M. Abdel-Basset, R. Mohamed, and S. Mirjalili, "A novel Whale Optimization Algorithm integrated with Nelder–Mead simplex for multi-objective optimization problems," *Knowl. Based Syst.*, vol. 212, p. 106619, 2021. [CrossRef]
18. D. Izci, S. Ekinci, M. Kayri, and E. Eker, "A novel enhanced metaheuristic algorithm for automobile cruise control system," *Electrica*, vol. 21, no. 3, pp. 283–297, 2021. [CrossRef]
19. D. Izci, S. Ekinci, S. Orenc, and A. Demiroren, "Improved artificial electric field algorithm using Nelder-Mead simplex method for optimization problems," in 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), 2020, pp. 1–5. [CrossRef]
20. D. Izci, and S. Ekinci, "A novel hybrid ASO-NM algorithm and its application to automobile cruise control system," 1st ed., in 2nd International Conference on Artificial Intelligence: Advances and Applications, G. Mathur, M. Bundele, L. Mahendra, and M. Paprzycki, Eds. Singapore: Springer, 2022.
21. D. Izci, S. Ekinci, H. L. Zeynelgil, and J. Hedley, "Performance evaluation of a novel improved slime mould algorithm for direct current motor and automatic voltage regulator systems," *Trans. Inst. Meas. Control*, vol. 44, no. 2, pp. 435–456, 2022. [CrossRef]
22. D. Izci, B. Hekimoğlu, and S. Ekinci, "A new artificial ecosystem-based optimization integrated with Nelder-Mead method for PID controller design of buck converter," *Alex. Eng. J.*, vol. 61, no. 3, pp. 2030–2044, 2022. [CrossRef]
23. N. H. Awad, M. Z. Ali, J. Liang, B. Y. Qu, and P. N. Suganthan, "Problem definitions and evaluation criteria for the CEC 2017 special session and competition on real-parameter optimization," *Nanyang Technol. Univ., Singapore, Tech. Rep.*, 2016, pp. 1–34.
24. Q. Askari, M. Saeed, and I. Younas, "Heap-based optimizer inspired by corporate rank hierarchy for global optimization," *Expert Syst. Appl.*, vol. 161, p. 113702, 2020. [CrossRef]
25. S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014. [CrossRef]
26. F. A. Hashim, K. Hussain, E. H. Houssein, M. S. Mabrouk, and W. Al-Atabany, "Archimedes optimization algorithm: A new metaheuristic algorithm for solving optimization problems," *Appl. Intell.*, vol. 51, no. 3, pp. 1531–1551, 2021. [CrossRef]
27. A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Futur Gener. Comput. Syst.*, vol. 97, pp. 849–872, 2019. [CrossRef]
28. D. Izci, S. Ekinci, A. Demiroren, and J. Hedley, "HHO Algorithm based PID Controller Design for Aircraft Pitch Angle Control System," in International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), 2020, pp. 1–6. [CrossRef]

Murat Kayri was graduated from Gazi University in Computer Science in 1994. He received his Msc from Yuzuncu Yil University in Electrical and Electronic Engineering and his PhD from Biometry and Genetics. He is currently a Professor at Yuzuncu Yil University and working on Artificial intelligence and control technologies.

Cengiz İpek graduated from Yüzüncü Yıl University, Faculty of Education, Department of Elementary Mathematics in 2002. He completed his master's degree with thesis in the Department of Statistics at Yüzüncü Yıl University in 2021. He is currently working at Van İpek Yolu Cumhuriyet Secondary School.

Davut İzci received his BSc degree from Dicle University, Turkey, in Electrical and Electronic Engineering and his MSc and PhD degrees from Newcastle University, England - UK, in Mechatronics and Microsystems, respectively. He is currently an Assistant Professor working at Batman University, Turkey. He carries out teaching and research activities both in Faculty of Engineering and Vocational School of Technical Sciences. His research interests are in microsystems development, sensing applications, robotics, and the applications of metaheuristic optimization algorithms to various control systems.

Erdal Eker was born in Van, Turkey in 1973. He was graduated from Van Yuzuncu Yil University, Turkey, in Mathematics. He received his Msc from Ataturk University, Turkey, in Applied Mathematics and received his PhD from Yuzuncu Yil University in Statistics. He is currently an instructor at Mus Alparslan University, Turkey