STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **15**, March 2026, pp 1791–1809.
Published online in International Academic Press (www.IAPress.org)

Trigonometric Functions Algorithm : A Novel Metaheuristic Algorithm for Engineering Problems

Wendinda BAMOGO

Department of Mathematics, Thomas SANKARA University, Burkina Faso

Abstract This paper deals with the design of a novel metaheuristic algorithm called Trigonometric Functions Algorithm (TFA) for efficient solving of engineering problems. The fundamental inspiration for this new algorithm is based on a mathematical model inspired by the hunting and attack technique of grey wolves and using trigonometric functions. For better exploration and exploitation of the search space, several random and adaptive variables are used. The various stages of well-arranged TFA are described and mathematically modeled. In order to prove the effectiveness and robustness of TFA, many engineering optimization problems of different difficulties were solved and a statistical study was made. The optimization results obtained with TFA were compared with the results of other state-of-the-art algorithms. Statistical and comparative studies showed that TFA achieves the best results and generally ranks first among the solved problems. The study of the sensitivity of TFA related to several parameters shows that TFA has a high degree of stability giving it the ability to efficiently solve optimization problems. In summary, the various studies have highlighted the efficiency, robustness and superiority of TFA compared to other competing algorithms and thus allow us to conclude that TFA remains a better option for solving technical design optimization problems.

Keywords Optimization, engineering problems, metaheuristics, TFA.

AMS 2010 subject classifications 90C15, 90C26, 90C30.

DOI: 10.19139/soic-2310-5070-3071

1. Introduction

In a fast-growing and competitive world, decision-makers, inventors and engineers urgently need efficient, robust and effective optimization methods that can effectively solve optimization problems for a rational use of raw materials and/or resources in all sectors for sustainable development. Several metaheuristic methods inspired by several phenomena exist in the literature for solving optimization problems.
Some are inspired by natural phenomena such as evolutionary algorithms such as : Genetic Algorithm (GA)[1], Differential Evolution (DE)[2], etc.
Others are inspired by swarm intelligence such as the behavior of certain animals, such as : Particle Swarm Optimization (PSO)[3], Predators Attacks Techniques Algorithm (PATA)[4], Zebra Optimization Algorithm (ZOA)[5], Tasmanian Devil Optimization (TDO)[6], Northern Goshawk Optimization (NGO)[7], Dung Beetle Optimizer (DBO)[8], Golden Jackal Optimization (GJO)[9], The Whale Optimization Algorithm (WOA)[10], Grey Wolf Optimizer (GWO)[11],etc, or certain behaviors of men such as : a new human-based metaheuristic algorithm for solving optimization problems on the base of simulation of driving training process (DTBO)[12], a new human-inspired metaheuristic algorithm for solving optimization problems based on mimicking sewing training (STBO)[13], etc.

*Correspondence to: Wendinda BAMOGO (Email: wdindabamogo@gmail.com). Department of Mathematics, Thomas SANKARA University, 12 BP 417 Ouagadougou 12, Saaba, Burkina Faso.

ISSN 2310-5070 (online) ISSN 2311-004X (print)

Other metaheuristics are inspired by physical laws such as : a Gravitational Search Algorithm (GSA)[14], a Multi-Verse Optimizer (MVO)[15], etc, and other metaheuristics by mathematical models such as : The Arithmetic Optimization Algorithm (AOA)[16], a new optimization algorithm based on Average and Subtraction of the Best and worst members of the population for solving various Optimization problems (ASBO)[17], a Sine Cosine Algorithm (SCA)[18], etc.
Despite this multitude of methods, most of them have numerous shortcomings. Most existing metaheuristics suffer from poor exploration and/or exploitation of the search space [10, 15, 16, 17, 18]. Generally, when a metaheuristic is rich in exploration, it is poor in exploitation and vice versa. This does not give them the ability to avoid local optima in favor of global optima.
Thus, to somewhat overcome this shortcoming that metaheuristics generally encounter, we propose in this work a new metaheuristic called Trigonometric Functions Algorithm (TFA) capable of balancing the exploration and exploitation phases for an efficient resolution of technical design optimization problems. The different stages of the TFA algorithm were designed by taking inspiration from the hunting and attack technique of gray wolves and using trigonometric functions. The arrangement and consistency of the different stages of TFA allow it to properly explore and exploit the search space and avoid local optima in favor of global optima.
The main innovative contributions of TFA are as follows :
-Maintaining a well-diversified population of solutions through controlled and intelligent perturbation of search agents by trigonometric functions during the optimization process,
-Avoiding premature convergence through a nonlinear convergence control parameter,
-The gradual transition from the exploration phase to the exploitation phase due to the convergence control parameter and the properties of trigonometric functions,
Avoiding local optima due to the oscillatory movements of the cosine and sine functions and the unbounded nature of the tangent function.
To highlight the effectiveness, robustness and performance of TFA, several engineering design optimization problems of different difficulties were successfully solved. A comparative and statistical study was carried out to prove the performance and superiority of TFA in the field of optimization.
For a clear understanding of this work, the document is structured as follows: after an introduction to the section (1), section (2) is first devoted to describing the proposed new metaheuristic, then the optimization results and performance study of TFA are presented in section (3), and finally we conclude with a conclusion and outlook in section (4).

2. Trigonometric Functions Algorithm

In this section, the different steps of the proposed new metaheuristic are presented.

2.1. Inspiration

The Trigonometric Functions Algorithm (TFA) metaheuristic is designed using a mathematical model that partially mimics the hunting and attacking techniques of grey wolves in nature and uses trigonometric functions [11, 18].

2.2. Mathematical modeling

In this subsection, mathematical models modeling the search, hunting and attack techniques of grey wolves and using trigonometric functions are presented in order to design the TFA algorithm.

2.2.1. INITIALIZATION Grey wolves are predators that generally hunt in packs and randomly search for prey within their hunting territory [4, 5, 11]. The following mathematical model :

$$X = lb + rand(N, d) \times (ub - lb) \tag{1}$$

allows us to obtain a population of solutions of the form below :

$$X = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_N \end{bmatrix} = \begin{bmatrix} x_{1,1} & x_{1,2} & \dots & x_{1,d} \\ x_{2,1} & x_{2,2} & \dots & x_{2,d} \\ \vdots & \dots & \dots & \vdots \\ x_{N,1} & x_{N,2} & \dots & x_{N,d} \end{bmatrix} \tag{2}$$

modeling the grey wolf population.
with lb, ub indicating the limits of the search space and N, d representing respectively the size of the wolf population and the dimension of the problem to be solved.

2.2.2. PERFORMANCE EVALUATION In order to determine the best wolf during the hunt, the performance of the different wolves must be evaluated[5, 11]. Thus, a performance evaluation of each solution is made via the function to be optimized in the following form :

$$F(X_i) = \begin{bmatrix} F(X_1) \\ F(X_2) \\ \vdots \\ F(X_N) \end{bmatrix} \tag{3}$$

in order to determine the best solution, denoted X_{best}, and save it.
When the problem to be optimized is to be maximized, X_{best} is the solution that gave the highest value of the function to be optimized and the lowest value of the function to be optimized, if the problem is to be minimized.

2.2.3. EXPLORATION AND EXPLOITATION PHASES Generally, exploration of the search space and hunting are guided by a leader wolf, which is not necessarily the best wolf[11]. Thus, TFA uses the mathematical models below, where model (4) models the exploration phase and models (6) and (7) model the exploitation phase.
Since the leader wolf is not necessarily the best wolf, it is chosen as follows: for each search agent, a random integer vector K of length $N-1$ is generated in the interval $[1, N]$ and the agent occupying the position of the first component of the vector K is always chosen. Depending on the performance of the search agent relative to the leader agent, the search agent adapts its position using the following mathematical model (4) :

$$X_i(t+1) = \begin{cases} X_i(t) + rand().(X_{leader}(t) - X_i(t)) \; if \; f(X_i(t)) > f(X_{leader}(t)), \\ X_i(t) + rand().(X_i(t) - X_{leader}(t)), \text{ else} \end{cases} \tag{4}$$

$$\begin{cases} a(t) = 2\Big(1 - \sqrt{\dfrac{t}{T}}\Big) \\ A_i(t) = 2.a(t).rand() - a(t), \; i \in \{1, 2, 3\} \end{cases} \tag{5}$$

$$\begin{cases} X_1(t) = X_i(t) + \cos(A_1(t)).\Big(X_{best}(t) - X_j(t)\Big) \\ X_2(t) = X_i(t) + \sin(A_2(t)).\Big(X_{best}(t) - X_j(t)\Big) \\ X_3(t) = X_i(t) + \tan(A_3(t)).\Big(X_{best}(t) - X_j)(t)\Big) \end{cases} \tag{6}$$

A priori, the best position sought, which is the position of the prey, is not known in advance, so an average position is used in the mathematical model (7) for this purpose.

$$X_i(t+1) = \frac{X_1(t) + X_2(t) + X_3(t)}{3}, \; i \in \{1, \dots, N\} \tag{7}$$

where $a(t)$, depending on t, is a coefficient that decreases from 2 to 0. T indicates the maximum number of iterations and t indicates the number of the current iteration. $rand()$ is a function that randomly generates a number in $[0, 1]$.

$X_i(t)$, $X_{leader}(t)$, $X_{best}(t)$ and $X_j(t),\ i, j \in \{1, \ldots, N\}$ represent respectively the current position of wolf i, the current position of the leader wolf, the position of the best wolf, and a randomly chosen position of wolf j.
In mathematical model (6), trigonometric functions are used for the following reasons :
-The oscillatory and bounded nature of the cosine and sine functions allows other candidate solutions to oscillate around the best solution, which leads to a global exploration of the space and prevents getting stuck in local minimum areas. It also allows for better exploitation in order to improve the quality of the best solution.
-The unbounded nature of the tangent function also allows, on the one hand, a large search space to be covered, which leads to better exploration of the space and, on the other hand, the tangent having very steep slopes around $\pm\frac{\pi}{2}$ leads to unpredictable and rapid movements of the search agents, which promotes stochastic behavior, which is very necessary to combat stagnation.

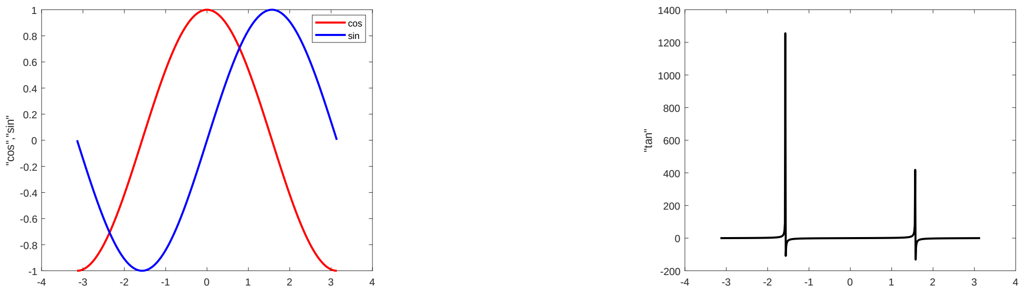


Figure 1. Left : Cosine and Sine functions curves in $[-\pi, +\pi]$; Right : Tangente function curve in $[-\pi, +\pi]$

-As the number of iterations increases and draws to a close, the amplitudes of the trigonometric functions decrease, promoting a gradual transition from the exploration phase to the exploitation phase.

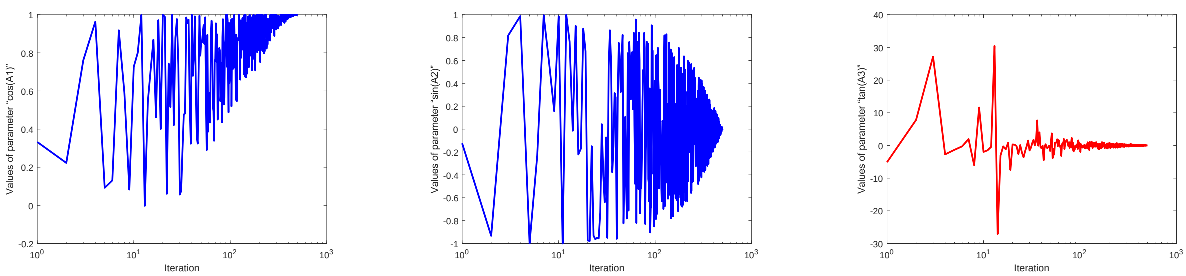


Figure 2. Behavior of the parameters cos(A1), sin(A2), and tan(A3) during the optimization process.

2.2.4. CONSTRAINT MANAGEMENT TECHNIQUE Although there are several constraint management techniques in the literature, constraint management in TFA is similar to that in the metaheuristics PATA[4], ZOA[5], and GWO[11] and is as follows :
For an optimization problem of the following form :

$$\begin{cases} \min f(x) \\ g_i(x) \leq 0 \; i = 1, \ldots, m \\ h_j(x) = 0 \; i = 1, \ldots, p \\ x \in \mathbb{R}^n \end{cases} \tag{8}$$

Constraints are penalized as follows :

$$\rho_1 \sum_{i=1}^{m} \max(0, g_i(x))^2 + \rho_2 \sum_{i=1}^{p} |h_j(x)| \tag{9}$$

where ρ_1, ρ_2 are penalty coefficients generally chosen empirically such that the more the constraint is violated, the more the solution is penalized.
Thus, the optimization problem becomes :

$$\begin{cases} \min f(x) + \rho_1 \sum_{i=1}^{m} \max(0, g_i(x))^2 + \rho_2 \sum_{i=1}^{p} |h_j(x)| \\ lb \leq x \leq ub \end{cases} \tag{10}$$

2.2.5. REPETITION PROCESS AND PSEUDO-CODE The TFA algorithm is an iterative algorithm with a population of solutions. At each iteration, it identifies the best general solution, saves it, and attempts to improve it as the number of iterations increases. At the end of the iterations, it returns the best solution as the optimal solution to the problem. The TFA algorithm is as follows :

Algorithm 1 Pseudo-code of TFA

Initialize the algorithm parameters (N, T, lb, ub, d, f),
Initialize the wolf population X with $X = lb + rand(N, d).(ub - lb)$.,
For $t = 1 : T$,
    For $i = 1 : N$,
        Calculate $F(X_i)$,
    end for,
    $X_{best}(t)$ assigns the best grey,
    $a(t) = 2\Big(1 - \sqrt{\frac{t}{T}}\Big)$,
    For $i = 1 : N$,
        Determine $X_{leader}(t)$,
        If $f(X_i(t)) > f(X_{leader}(t))$,
            $X_i(t+1) = X_i(t) + rand().(X_{leader}(t) - I.X_i(t))$,
        Else,
            $X_i(t+1) = X_i(t) + rand().(X_i(t) - X_{leader}(t))$,
        End
        $j = floor(rand() * N) + 1$;
        While $j == i$
            $j = floor(rand() * N) + 1$;
        End while
        $A_1(t) = 2.a(t).rand() - a(t)$
        $X_1(t) = X_i(t) + \cos(A_1(t)).(X_{best}(t) - X_j(t))$
        $A_2(t) = 2.a(t).rand() - a(t)$
        $X_2(t) = X_i(t) + \sin(A_2(t)).(X_{best}(t) - X_j(t))$
        $A_3(t) = 2.a(t).rand() - a(t)$,
        $X_3(t) = X_i(t) + \tan(A_3(t)).(X_{best}(t) - X_j)(t)$,
        $X_i(t+1) = \frac{X_1(t) + X_2(t) + X_3(t)}{3}$,
    End for,
End for,
Return $X_{best}(T)$ as the optimal solution,

2.2.6. COMPUTATIONAL COMPLEXITY In this subsection, an analysis of the complexity of TFA proposed in this work is performed. The complexity of TFA depends heavily on three main steps: the initialization process of the solution population, the evaluation of solution performance, and the updating of solutions.
The technique used to initialize the solution population in TFA is similar to that used in the nine other comparative algorithms in this work, which has a complexity equal to $O(N.d)$ [5, 6, 7].

In one iteration:
- The process of evaluating the performance of solutions is given by $O(N.c)$ where c is the cost of evaluation per solution. Since the cost of evaluation depends heavily on the dimension d of the problem, this complexity can be approximated by $O(N.d)$.
- The complexity of determining the best solution is given by $O(N)$.
Thus, for all iterations, the total complexity of the evaluation is given by $O(N.T.d)$ and that of determining the best solution by $O(N.T)$.
During the optimization process, each member of the solution population is updated in two phases, each of which has a complexity equal to $O(N.T.d)$ and its performance is evaluated[5, 6, 7].
Thus, the total complexity of updating search agents is equal to $O(2.N.T.d)$. This gives TFA a total complexity of $O(N.d) + O(N.T) + O(N.T.d) + O(2.N.T.d)$, which is approximately equal to $O(N.T(2 + 3.d))$.

3. Results and Discussions

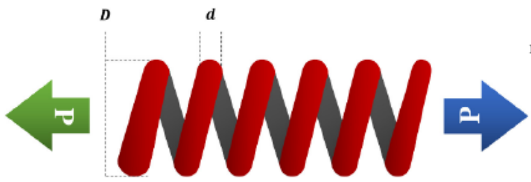
In this section, the ability of the proposed TFA algorithm to solve engineering optimization problems is implemented on four technical design challenges, such as tension/compression spring design, pressure vessel design, welded beam design, and speed reducer design, which are all minimization problems [5, 6, 7, 9, 16].

3.1. Simulation studies

The performance of the proposed algorithm is compared to the performance of twelve other state-of-the-art metaheuristics such as ZOA, TDO, NGO, AOA, DBO, GJO, ASBO, WOA, GWO, PSO, DE and GA.
The proposed TFA algorithm and each of the algorithms mentioned are executed in thirty independent implementations, each execution comprising five hundred iterations.
The general parameters for all algorithms are the size of the solution population N, the maximum number of iterations T, the problem dimension d, and the search space $[lb, ub]$. The remaining parameters are specific to each algorithm.
The implementations are performed in the MATLAB R2020a environment with a 64-bit operating system computer, x64 processor Intel(R) Core(TM) i5-5300U CPU @ 2.30GHz with 8.00GB RAM.
The optimization results for each problem were reported in a table using five indicators: the best optimal solution (Best), the average of the optimal solutions obtained (Mean), the standard deviation of these solutions (Std), the median (Median), and the worst solution (Worst).

3.1.1. Tension/compression spring design problems The design of a tension/compression spring is an optimization problem whose main objective is to reduce the weight of the spring [4, 5, 16].
Figure 3 presents the scheme and mathematical formulation of this problem. The statistical results of the optimization are reported in Table 1. The convergence curves and boxplot of performance of TFA and other competing algorithms are shown in Figure 4.

3.1.2. Pressure vessel design problem The pressure vessel design problem is an engineering optimization problem with the objective to evaluate the optimal thickness of shell $(Ts = x_1)$, thickness of head $(Th = x_2)$, inner radius $(R = x_3)$, and length of shell $(L = x_4)$ such that the total cost of material, forming, and welding is minimized accounting for constraints [4, 5, 16].
Figure 5 presents the scheme and mathematical formulation of the problem. The statistical results are presented in Table 2 and the convergence and performance curves in Figure 6.



$$\min \quad f(x) = (x_3 + 2)x_2 x_1^2 \ with \ x_1 = d, x_2 = D, x_3 = P$$

$$sc: \begin{cases} g_1(x) = 1 - (x_3 x_2^3 / 71785 x_1^4); \\ gtmp = (4x_2^2 - x_1 x_2)/(12566(x_2 x_1^3 - x_1^4)) \\ g_2(x) = gtmp + 1/(5108 x_1^2) - 1; \\ g_3(x) = 1 - (140.45 x_1 / x_2^2 x_3); \\ g_4(x) = ((x_1 + x_2)/1.5) - 1; \\ 0.05 \le x_1 \le 2; 0.25 \le x_2 \le 1.3; 2 \le x_3 \le 15 \end{cases}$$

Figure 3. Schematic view and mathematical formulation of the tension/compression spring design problem [5, 16].

Table 1. Statistical results for the tension/compression spring design problem

Algorithm					
	Best	Mean	std	Median	Worst
TFA	**0.012665232788319**	**0.012665236984553**	**0.000000015919007**	**0.012665232851741**	**0.012665318514362**
ZOA	0.012668500848822	0.013046077154204	0.000329487689804	0.012942205963493	0.013979832422859
TDO	0.012666551401355	0.012676998639520	0.000010717640638	0.012676399993037	0.012727450215429
NGO	0.012665997501453	0.012675079496948	0.000011876766097	0.012671000047615	0.012731829340704
AOA	0.013197598316181	0.019135331215311	0.011590980394300	0.013239713565894	0.064189096595329
DBO	0.012719053710187	0.013681351919925	0.001733745282979	0.013049424335061	0.018479548067360
GJO	0.012698961077449	0.012852143939329	0.000195674466581	0.012761440963046	0.013612271154547
ASBO	0.012745224527006	0.013082872996868	0.000238267051120	0.013059802085022	0.013820793060232
WOA	0.012665744420485	0.013650874881519	0.001321371447425	0.013087086394657	0.017773933021309
GWO	0.012671644908351	0.012801036663875	0.000204330370293	0.012737980836286	0.013674443300573
PSO	0.012670259812914	0.013427338904796	0.001074131764219	0.013063440797390	0.017694045750243
DE	0.012707957806864	0.012931050849562	0.000236317086298	0.012854634908652	0.013518467259336
GA	0.012675733619420	0.012900518707469	0.000163257070609	0.012873842733649	0.013335454109050

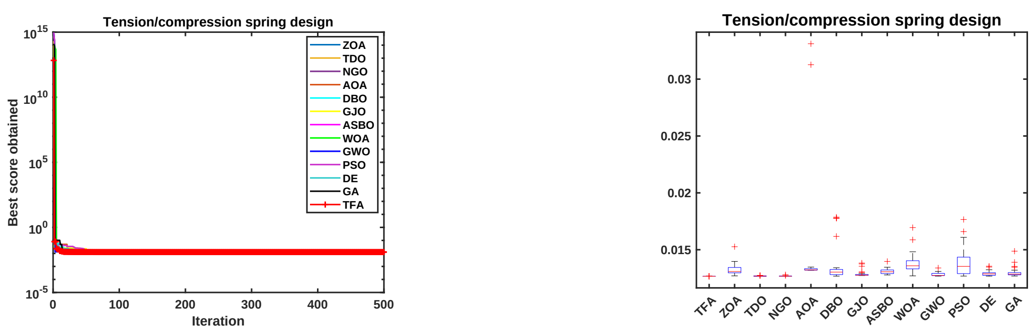
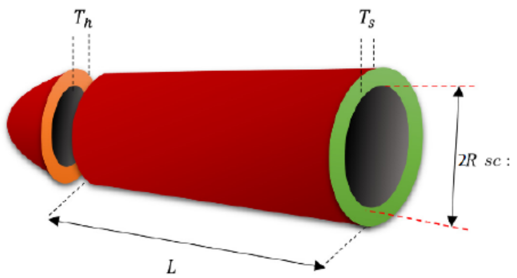


Figure 4. Left : Convergence curves of TFA and competitor algorithms; Right : Boxplot of performance of TFA and competitor algorithms



$$\min \quad f(x) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3 \; with \; x_1 = T_s,\; x_2 = T_h,\; x_3 = R,\; x_4 = L,$$

$$sc: \begin{cases} g_1(x) = -x_1 + 0.0193x_3; \\ g_2(x) = -x_2 + 0.00954x_3; \\ g_3(3x) = -\pi x_3^2 x_4 - (4/3)\pi x_3^3 + 1296000; \\ g_4(x) = x_4 - 240; \\ 0 \leq x_1, x_2 \leq 99; 10 \leq x_3, x_4 \leq 200 \end{cases}$$

Figure 5. Schematic view and mathematical formulation of the pressure vessel design problem[5, 16].

Table 2. Statistical results for the pressure vessel design problem

Algorithm	Best	Mean	standard deviation	Median	Worst
TFA	**5885.332773601228**	**5885.332773601228**	**0**	**5885.332773601228**	**5885.332773601228**
ZOA	6.548.665684409157	7.383.300672760658	0.465.430353326781	7.307.932816930937	8.295.875051716388
TDO	5885.338672313107	5911.200355786549	46.747557586522	5891.508636500136	6109.574202349156
NGO	5885.332788746427	5924.368149621047	64.476211622026	5897.925153977478	6154.091271137160
AOA	7963.3997586812	47603.1560038019	38864.7442317307	34115.5617692369	156155.6856961688
DBO	5885.332773601228	6457.074150143470	646.117139394803	5957.231297260766	7319.000702032429
GJO	5910.376878268812	6303.991541703866	525.585775290289	5995.188169868685	7306.793269404162
ASBO	6976.5020978769	21591.1120301172	41285.3372480498	9580.2129616768	226007.3971110758
WOA	6513.94170155727	11550.06742029256	8145.38254335810	8683.26347685146	46940.92716292769
GWO	5893.280815465118	6214.751032721023	463.073156951266	5957.636049150820	7218.287340577674
PSO	5919.397948599343	6274.306270868351	294.235154209288	6160.543610928559	6999.581112727349
DE	5895.752036076108	6225.454123614910	253.952005185361	6176.927517177426	6753.337733001699
GA	5933.174354969110	6243.308277486108	280.321490521061	6209.553651448397	7319.000702032429

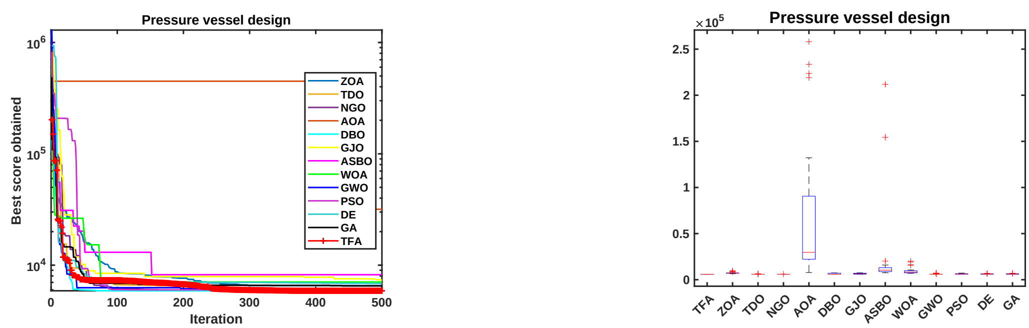
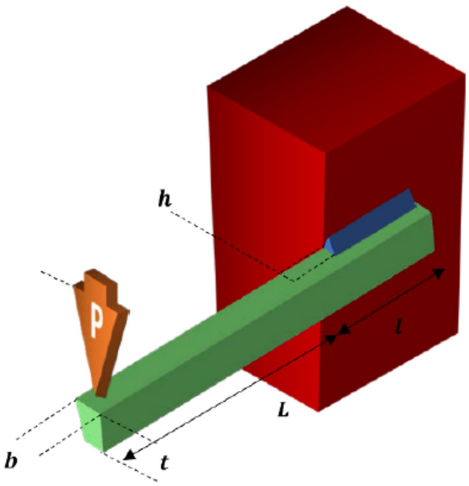


Figure 6. Left : Convergence curves of TFA and competitor algorithms; Right : Boxplot of performance of TFA and competitor algorithms

3.1.3. Welded beam design problem The welded beam is a common engineering optimisation problem with an objective to find an optimal set of the dimensions $h = x_1, l = x_2, t = x_3$, and $b = x_4$ such that the fabrication cost of the beam is minimized [4, 5, 16].

The scheme and mathematical formulation of this design problem are presented in Figure 7. In Table 3 are presented the statistical results of the optimization and the convergence and performance curves in Figure 8.



$$\min \quad f(x) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14.0 + x_2);$$

$$with\ x_1 = h,\ x_2 = l,\ x_3 = t,\ x_4 = b,$$

$$sc: \begin{cases} Q = 6000(14 + x_2/2); \\ D = \sqrt{(x_2^2/4 + (x_1 + x_3)^2/4)}; \\ J = 2(x_1x_2\sqrt{2}(x_2^2/12 + (x_1 + x_3)^2/4)); \\ \alpha = 6000/\sqrt{2}x_1x_2; \\ \beta = QD/J; \\ \tau = \sqrt{(\alpha^2 + 2\alpha\beta x_2/2D) + \beta^2)}; \\ \sigma = 504000/x_4x_3^2; \\ \delta = 65856000/30.10^6x_4x_3^3; \\ F = 4.013(30.10^6)/196\sqrt{(x_3^2x_4^6/36)}(1 - x_3\sqrt{(30/48)}/28); \\ g_1(x) = \tau - 13600; \\ g_2(x) = \sigma - 30000; \\ g_3(x) = x_1 - x_4; \\ g_4(x) = 0.10471x_1^2 + 0.04811x_3x_4(14 + x_2) - 5.0; \\ g_5(x) = 0.125 - x_1; \\ g_6(x) = \delta - 0.25; \\ g_7(x) = 6000 - F; \\ 0.1 \leq x_1, x_4 \leq 2; 0.1 \leq x_2, x_3 \leq 10 \end{cases}$$

Figure 7. Schematic view and mathematical formulation of the welded beam design problem[5, 16].

Table 3. Statistical results for the welded beam design problem

Algorithm					
	Best	Mean	standard deviation	Median	Worst
TFA	**1.724852308597364**	**1.724852308597363**	**0.000000000000001**	**1.724852308597364**	**1.724852308597364**
ZOA	1.725395156190328	1.900142840951764	0.155664207477050	1.910586504048673	2.226296255556301
TDO	1.724852308899247	1.724852336991362	0.000000064728449	1.724852315330008	1.724852653703401
NGO	1.724852308975615	1.724852548350067	0.000000963314587	1.724852323045800	1.724857603183851
AOA	1.947181200782214	2.499607469346151	0.338093773639409	2.516309528163525	3.262054473581157
DBO	1.724856929249184	1.767462431921528	0.058941320807906	1.733775475095067	1.993433445517653
GJO	1.729050142557291	1.736763570894445	0.006619715974791	1.734682331280755	1.754556987880942
ASBO	1.815033721397199	2.974174322151296	0.730109160331940	3.206198528990591	4.014796952108314
WOA	1.886878530963351	2.923232551057374	0.937322945279733	2.689217347343203	6.345665795446908
GWO	1.727043578162027	1.729605808805834	0.002360413997967	1.728984302918515	1.738569928936277
PSO	1.724852313712982	1.832217163481421	0.237000293812207	1.725838954769771	2.945963721748117
DE	1.794593321903436	2.049950215908519	0.159185780686323	2.046866687516128	2.416617923903347
GA	1.774212852288099	2.029510805152706	0.191555312072770	1.998470176201013	2.637868265883140

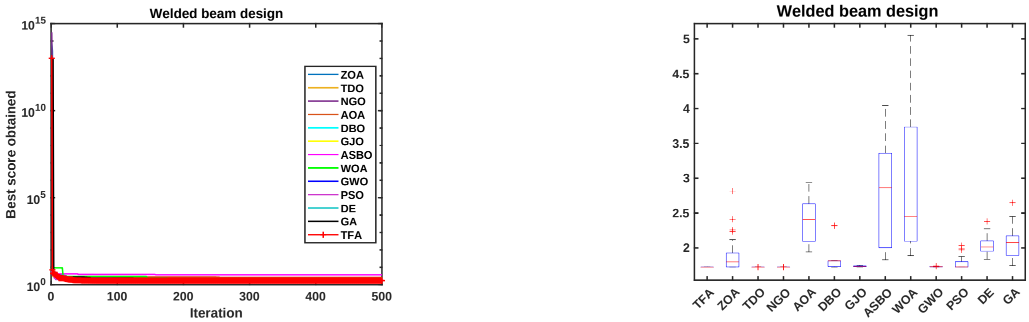
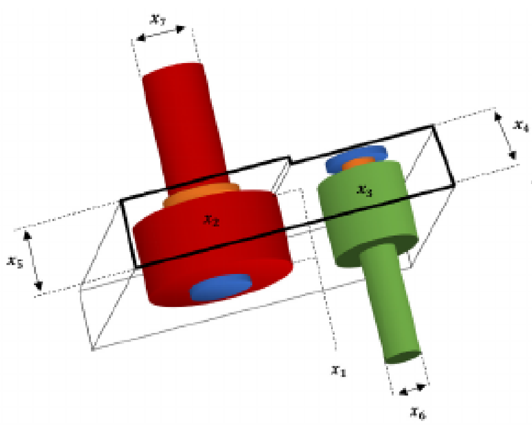


Figure 8. Left : Convergence curves of TFA and competitor algorithms; Right : Boxplot of performance of TFA and competitor algorithms

3.1.4. Speed reducer design problem The objective is to minimize the total weight of the speed reducer. There are nine constraints, including limits on gear tooth bending stress, surface stress, transverse deformations of shafts 1 and 2 due to the transmitted force, and stresses in shafts 1 and 2 [4, 5, 16].
The scheme and mathematical formulation of this problem are shown in Figure 9. The statistical results of solving this problem are shown in Table 4 and the curves illustrating the convergence and performance of TFA and the other algorithms are shown in Figure 10.



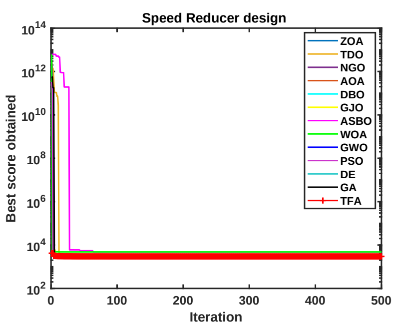
$$\min \quad f(x) = 0.7854x_1x_2^2(3.3333x_3^2 + 14.9334x_3 - 43.0934) - 1.508x_1(x_6^2 + x_7^2) + 7.4777(x_6^3 + x_7^3) + 0.7854(x_4x_6^2 + x_5x_7^2)$$

$$sc: \begin{cases} g_1(x) = (27/x_1x_2^2x_3) - 1; \\ g_2(x) = (397.5/x_1x_2^2x_3^2) - 1; \\ g_3(x) = (1.93x_4^3/x_2x_6^4x_3) - 1; \\ g_4(x) = (1.93x_5^3/x_2x_7^4x_3) - 1; \\ g_5(x) = (1/110x_6^3)\sqrt{(745x_4/x_2x_3)^2 + 16.9.10^6} - 1; \\ g_6(x) = (1/85x_7^3)\sqrt{(745x_5/x_2x_3)^2 + 157.5.10^6} - 1; \\ g_7(x) = (x_2x_3/40) - 1; \\ g_8(x) = (5x_2/x_1) - 1; \\ g_9(x) = (x_1/12x_2) - 1; \\ g_{10}(x) = ((1.5x_6 + 1.9)/x_4) - 1; \\ g_{11}(x) = ((1.1x_7 + 1.9)/x_5) - 1; \\ 2.6 \le x_1 \le 3.6; 0.7 \le x_2 \le 0.8; 17 \le x_3 \le 28; 7.3 \le x_4 \le 8.3; \\ 7.3 \le x_5 \le 8.3; 2.9 \le x_6 \le 3.9; 5.0 \le x_7 \le 5.5 \end{cases}$$

Figure 9. Schematic view and mathematical formulation of the speed reducer design problem[5, 16].

Table 4. Statistical results for the speed reducer design problem

Algorithm					
	Best	Mean	standard deviation	Median	Worst
TFA	**2994.471066145981**	**2994.488256163399**	**0.094152862524**	**2994.471066145982**	**2994.986762747748**
ZOA	2.997293349555451	3.004535876641060	0.005162614033988	3.003369676367997	3.017375713695672
TDO	2994.471066146066	2994.471066164749	0.000000026884	2994.471066153120	2994.471066256889
NGO	2994.471066146025	2994.471066166476	0.000000020843	2994.471066165526	2994.471066236879
AOA	3064.905818703883	3185.227573348073	37.424012933474	3196.790253658783	3234.228434730951
DBO	2994.471066145981	3042.830602783940	57.097963158813	3033.748525701569	3202.727133950403
GJO	3010.112519133358	3026.111375716775	8.706783240001	3024.806311134280	3051.052806756329
ASBO	3131.051525152855	4520.544847900663	757.401897082661	4689.515559129180	5541.128153268632
WOA	3037.538738882842	3348.744895937827	587.922374160313	3154.559257986750	5406.149873821764
GWO	3003.440386802583	3012.316605575711	4.957469396078	3012.230009066500	3022.274391975748
PSO	2994.471066145981	2994.471066145983	0.000000000004	2994.471066145981	2994.471066146002
DE	2994.471066145981	2994.471066145983	0.000000000001	2994.471066145981	2994.471066145981
GA	2994.471066145981	2994.471066145983	0.000000000001	2994.471066145981	2994.471066145981



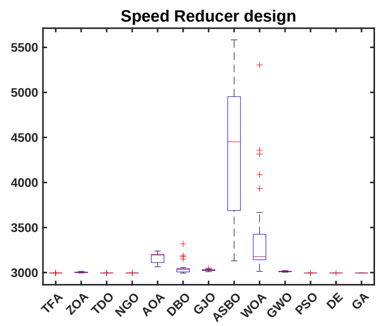


Figure 10. Left : Convergence curves of TFA and competitor algorithms; Right : Boxplot of performance of TFA and competitor algorithms

The results shown in Tables 1, 2, 3, and 4 above clearly demonstrate that the TFA metaheuristic outperforms other competing metaheuristics. TFA exhibits a higher degree of stability compared to other metaheuristics because, on average, it obtains the best objective function value and also the smallest std value. It also obtains the best possible results because, in the column of best results, we note that the results obtained by TFA are the best, which is interesting because all problems are to be minimized.

The curves in Figures 4, 6, 8 and 10 graphically and visibly illustrate the behavior and performance of TFA compared to other algorithms in determining optimal solutions to an optimization problem.
In summary, the statistical results of the simulations and the different curves show that TFA has a high performance and manages to provide the best optimal solutions on all the problems solved.

3.2. Sensitivity analysis of the TFA for N and T.

Since TFA is an iterative algorithm with a solution population, it is necessary to analyze its sensitivity with respect to these two parameters for the four problems solved.
Tables 5, 6, 7, and 8 contain the mean values and standard deviation of the results obtained for different values of N

and T.)
Curves 11, 12, 13, and 14 below show the sensitivity of TFA with respect to these two parameters with $T = 500$ and $N = 10, 50, 80, 100$ on the one hand and $N = 30$ and $T = 100, 500, 800, 1000$ on the other.

Table 5. TFA sensitivity analysis for the tension/compression spring design problem

N \ T		100	500	800	1000
10	mean	7603606678119.08	7603606678119.07	7603606678119.07	7603606678119.07
	std	41646668960027.35	41646668960027.35	41646668960027.36	41646668960027.36
50	mean	0.012665386767286	0.012665232829692	0.012665232788336	0.012665232788320
	std	0.000000655741389	0.000000000140960	0.000000000000050	0.000000000000002
80	mean	0.012665243712660	0.012665232788328	0.012665232788319	0.012665232788319
	std	0.000000013949539	0.000000000000043	0.000000000000000	0.000000000000000
100	mean	0.012665238731531	0.012665232788324	0.012665232788319	0.012665232788319
	std	0.000000009488043	0.000000000000021	0.000000000000000	0.000000000000000

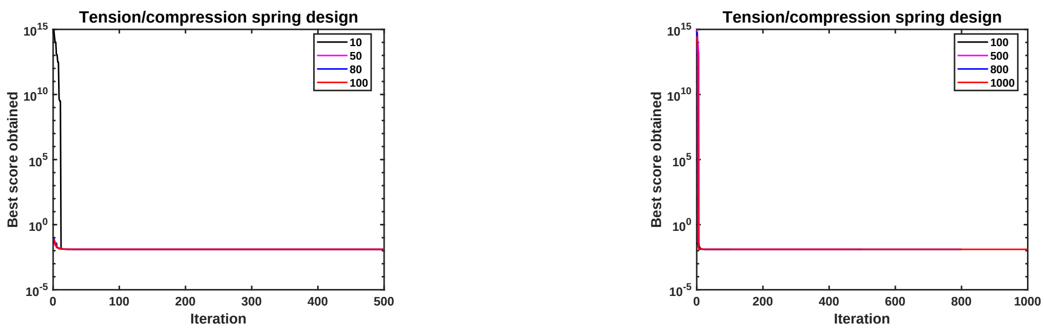


Figure 11. TFA sensitivity curve for the number of population members and the maximum number of iterations for the tension/compression spring design problem.

Table 6. TFA sensitivity analysis for the pressure vessel design problem

N \ T		100	500	800	1000
10	mean	6693.825134711047	6459.081732769483	6389.311853480958	6380.374550576166
	std	625.084461149458	469.610950462169	530.236599942498	486.319583067739
50	mean	5906.532226333673	5885.332773601228	5885.332773601228	5885.332773601228
	std	086.064736788831	0	0	0
80	mean	5885.343585790642	5885.332773601228	5885.332773601228	5885.332773601228
	std	0.006639475848	0	0	0
100	mean	5885.348901885253	5885.332773601228	5885.332773601228	5885.332773601228
	std	0.024579027038	0	0	0

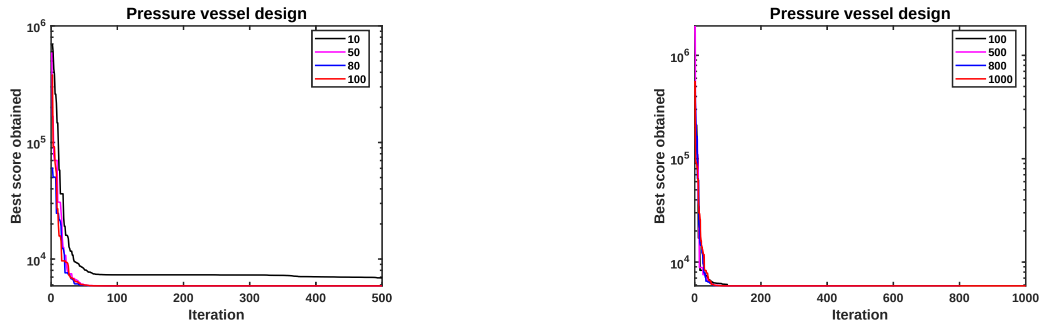


Figure 12. TFA sensitivity curve for the number of population members and the maximum number of iterations for the Pressure vessel design problem.

Table 7. TFA sensitivity analysis for the welded beam design problem

N \ T		100	500	800	1000
10	mean	2.027438418176958	1.904154440762991	1.948330332851710	1.834622862257717
	std	0.589843244310975	0.298607067122554	0.446939276851640	0.212855966432791
50	mean	1.724855612394897	1.724852308597363	1.724852308597363	1.724852308597363
	std	0.000002375768116	0.000000000000001	0.000000000000001	0.000000000000001
80	mean	1.724853718379197	1.724852308597363	1.724852308597363	1.724852308597363
	std	0.000000624453821	0.000000000000001	0.000000000000001	0.000000000000001
100	mean	1.724853534851678	1.724852308597363	1.724852308597363	1.724852308597363
	std	0.000000663899694	0.000000000000001	0.000000000000001	0.000000000000001

Table 8. TFA sensitivity analysis for the speed Reducer design problem

N \ T		100	500	800	1000
10	mean	3294.006431939565	3241.245965828405	3106.512673313132	3036.421214419665
	std	636.954964842616	568.323774671236	318.038567252345	53.469012707518
50	mean	2995.714988729958	2994.471066145983	2994.471066145983	2994.471066145983
	std	2.895069910000	0.000000000001	0.000000000001	0.000000000001
80	mean	2994.674162937013	2994.471066145983	2994.471066145982	2994.471066145983
	std	1.090283264778	0.000000000001	0.000000000001	0.000000000001
100	mean	2994.478262983979	2994.471066145983	2994.471066145983	2994.471066145983
	std	0.028327252303	0.000000000001	0.000000000001	0.000000000001

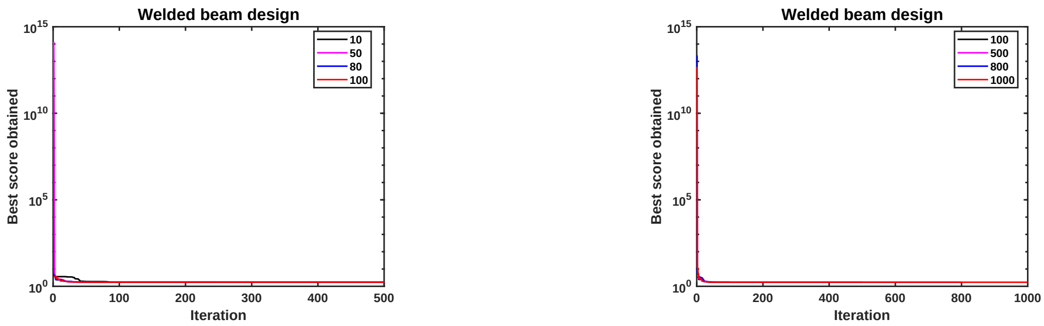


Figure 13. TFA sensitivity curve for the number of population members and the maximum number of iterations for the welded design problem.

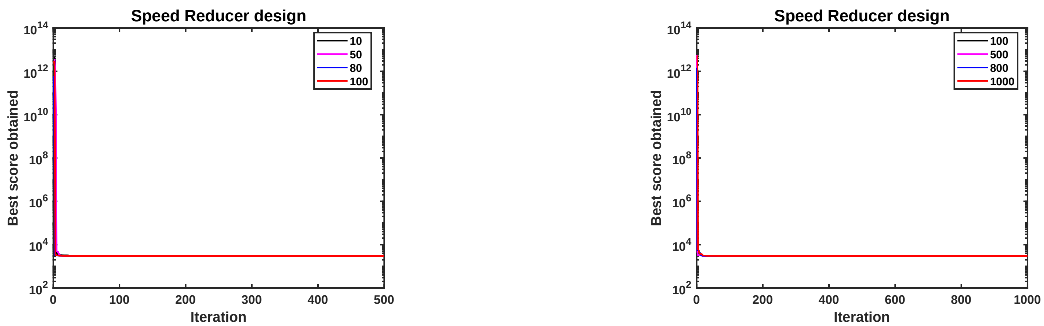


Figure 14. TFA sensitivity curve for the number of population members and the maximum number of iterations for the speed design problem.

Table 9 contains the results of the ANOVA statistical test in order to statistically visualize the sensitivity of TFA with respect to N and T.

Table 9. One-way anova test results.

Problems \ Parameters	N		T	
	F	p-value	F	p-value
Tension/compression spring design	1.52	0.2134	5.26	0.0019
Pressure vessel design	26.85	2.91658e-13	1.99	0.1194
Welded beam design	6.96	0.0002	5.46	0.0015
Speed Reducer design	11.06	7.05041e-08	8.21	5.28461e-05

The data in Tables 5, 6, 7, and 8 show a slight sensitivity of TFA when the population size N is small ($N = 10$), because for this value, the solutions found by TFA are significantly different from the other solutions.

Graphically, the observation is that the curves (in Figures 11, 12, 13 and 14 are almost superimposed on each other, indicating that TFA has low sensitivity to the size (N) of the solution population and the maximum number T of iterations. This confirms once again that TFA has a high degree of stability compared to other competing algorithms, enabling it to obtain better solutions.
The results of the one-way ANOVA test in Table 9 statistically confirm the above analysis, as some p-values are strictly less than 0.05.

3.3. Population diversity and exploration-exploitation phases curves of the TFA.

In this subsection, we visualize the evolution of the solution population and the ability of TFA to balance the exploration and exploitation phases during the optimization process.

The curves in Figures 15, 16, 17, and 18 clearly demonstrate these properties of TFA for the four problems solved.

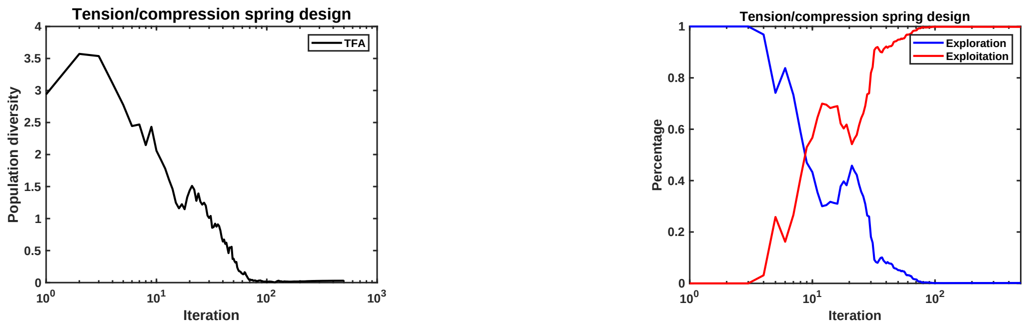


Figure 15. Curves showing population diversity and the exploration and exploitation phases of TFA for the tension/compression spring design problem.

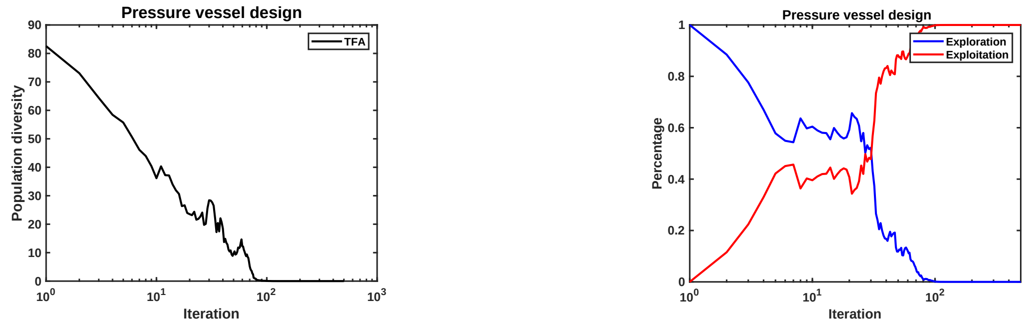


Figure 16. Curves showing population diversity and the exploration and exploitation phases of TFA for the pressure design problem.

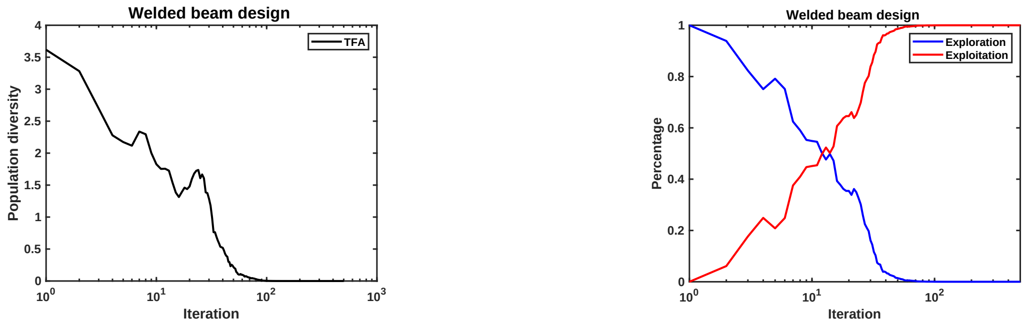


Figure 17. Curves showing population diversity and the exploration and exploitation phases of TFA for the welded design problem.

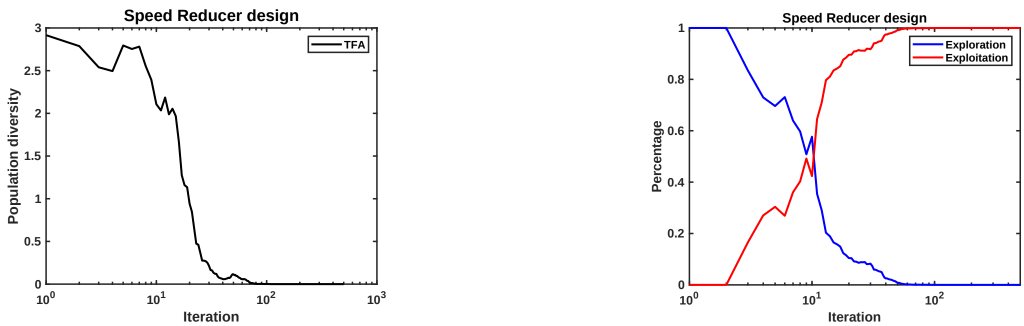


Figure 18. Curves showing population diversity and the exploration and exploitation phases of TFA for the speed design problem.

We clearly see that the degree of diversity of solutions decreases as the number of iterations increases, which is logical and characterizes the behavior of a good iterative algorithm with a solution population. In terms to the management of exploration and exploitation phases, we find that TFA manages to strike a balance between these two phases during the optimization process, enabling it to explore and exploit the search space appropriately and solve optimization problems efficiently.

3.4. Statistical analysis

In order to statistically prove the superiority of TFA over other competing algorithms, the Wilcoxon rank sum test, Friedman test, and tiedrank test are used for this purpose in this subsection [5, 6, 7].

The results of the various tests are recorded in Tables 10, 11, and 12 below.

Table 10. p-values obtained from Wilcoxon rank sum test.

Algorithms \ Problems	Tension	Pressure	Welded	Speed
TFA vs ZOA	3.019859359162151e-11	1.720251003131564e-12	1.211780397005987e-12	7.768636411654877e-12
TFA vs TDO	3.019859359162151e-11	1.944284567534306e-11	1.211780397005987e-12	3.035217048933032e-11
TFA vs NGO	3.019859359162151e-11	4.104006707056105e-11	1.211780397005987e-12	1.067590545413501e-11
TFA vs AOA	3.016075319890950e-11	1.720251003131564e-12	1.211780397005987e-12	7.768636411654877e-12
TFA vs DBO	3.000982378980077e-11	1.499284497878838e-08	1.211780397005987e-12	1.162801981623527e-10
TFA vs GJO	3.019859359162151e-11	8.159588532143750e-12	1.211780397005987e-12	7.768636411654877e-12
TFA vs ASBO	3.019859359162151e-11	1.720251003131564e-12	1.211780397005987e-12	7.768636411654877e-12
TFA vs WOA	3.019859359162151e-11	1.720251003131564e-12	1.211780397005987e-12	7.768636411654877e-12
TFA vs GWO	3.019859359162151e-11	2.409814666548896e-11	1.211780397005987e-12	7.768636411654877e-12
TFA vs PSO	3.019859359162151e-11	1.211780397005987e-12	1.720251003131564e-12	3.0512557802644e-02
TFA vs DE	3.019859359162151e-11	1.211780397005987e-12	1.720251003131564e-12	4.949666687210789e-06
TFA vs GA	3.019859359162151e-11	1.211780397005987e-12	1.720251003131564e-12	4.949666687210789e-06

Table 11. Friedman test results.

Compared algorithms	Mean.Rank	Overal Rank
TFA	1.25	1
ZOA	9.5	12
TDO	3.25	2
NGO	3.5	3
AOA	11.5	13
DBO	9.25	11
GJO	7.25	8
ASBO	9.5	12
WOA	9	10
GWO	5.75	4
PSO	8	9
DE	6.75	7
GA	6.5	5

Table 12. tiedrank test results.

Problems \ Algorithms	TFA	ZOA	TDO	NGO	AOA	DBO	GJO	ASBO	WOA	GWO	PSO	DE	GA
Tension/compression spring design	1	12	4	2	11	10	6	5	3	7	13	9	8
Pressure vessel design	1	7	2	3	10	11	9	12	13	4	8	6	5
Welded beam design	1	12	2	3	13	6	5	8	9	4	7	10	11
Speed Reducer design	2	7	5	6	12	10	9	13	11	8	4	2	2

The results of the Wilcoxon rank sum test confirm that the TFA algorithm proposed in this work is significantly superior to the competing algorithms used ($p < 5\%$). The results of the Friedman and Tiedrank tests show that the TFA algorithm is better and generally ranks first. Figure 19 illustrates this graphically.
In summary, the statistical optimization results, convergence curves, exploration and exploitation performance curves, and various statistical tests show that TFA has a high degree of stability and outperforms competing algorithms, making the TFA algorithm robust, efficient, and capable of effectively solving technical design optimization problems.

3.5. Convergence analysis

In this subsection, an analysis of the convergence behavior of the proposed TFA algorithm is performed. The curves in Figure 2 show that :

- At the beginning of the optimization process, the amplitudes of the parameters $\cos(A_1)$, $\sin(A_2)$ and $\tan(A_3)$ are large, causing sudden, large-step oscillatory movements by the search agents, which promotes in-depth exploration of promising regions of the search space.
- Towards the end of the optimization process, their amplitudes gradually decrease, resulting in reduced movement and allowing search agents to focus appropriately on exploitation.

According to Berg et al[10, 11, 19], an algorithm with a population of solutions exhibiting such behavior can guarantee that this algorithm will ultimately converge to a point in the search space.

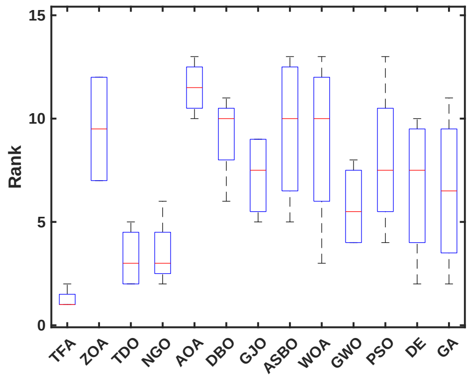


Figure 19. Boxplot of rank of TFA and competitor algorithms.

4. Conclusion

In this work, a new iterative algorithm with population of solutions operating on a mathematical model inspired by the hunting and attack technique of gray wolves and using trigonometric functions called Trigonometrics Functions Algorithm (TFA) is proposed. This algorithm has been successfully applied to solve several engineering design optimization problems. The optimization and performance study results have shown that TFA has good performance, enabling it to efficiently solve optimization problems with or without constraints. TFA remains a better alternative for efficient solving of practical or engineering design problems. Proposing the multi-objective version of TFA for efficient determination of Pareto optimal solutions of multi-objective engineering design optimization problems remains the subject of our future work.

REFERENCES

1. D. E. Goldberg and J. H. Holland, *netic algorithms and machine learning*, Mach. Learn., vol. 3, pp. 95–99, 1988.
2. R. Storn and K. Price, *Differential evolution a simple and efficient heuristic for global optimization over continuous spaces*, J. Global Optim., vol. 11, pp. 341–359, 1997.
3. J. Kennedy and R. Eberhart, *Particle swarm optimization*, in Proc. Int. Conf. Neural Netw., vol. 4, pp. 1942–1948, 1995.
4. W. Bamogo, and K. Some, *Predators Attacks Techniques Algorithm : a novel bio-inspired algorithm for angineering problems*, IEEE Multi-conference on Natural and Engineering Sciences for Sahel's Sustainable Development (MNE3SD), 28-30 November, Ouagadougou, Burkina Faso, pp. 95–100, 2024.

5. E. Trojovska, M. Dehghani ans P. Trojovský, *Zebra Optimization Algorithm : a new bio-inspired Optimization Algorithm for Solving Optimization Algorithm*, IEEE Acess, vol. 10, pp. 49445–49473, 2022.
6. M. Dehghani, S. Hubálovský and P. Trojovsky, *Tasmanian Devil Optimization : a new bio-inspired Optimization Algorithm for Solving Optimization Algorithm*, IEEE Acess, vol. 10, pp. 19599–19620, 2022.
7. M. Dehghani, Š. Hubálovský and P. Trojovský, *Northern Goshawk Optimization : a new swarm-based Algorithm for Solving Optimization Problems*, IEEE Acess, vol. 9, pp. 162059–162080, 2021.
8. J. Xue, B. Shen, *Dung Beetle Optimizer : a new meta-heuristic algorithm for global optimization*, the Journal of Supercomputing, vol. 9, pp. 7305–7336, 2023.
9. N. Chopra, M. Mohsin Ansari, *Golden Jackal Optimization : A novel nature-inspired optimizer for engineering applications*, Elsevier, Expert Systems with Applications, vol. 198, 2022.
10. S. Mirjalili, and A. Lewis, *The Whale Optimization Algorithm*, Advances in Engineering Software, vol. 95, pp. 51-67, 2016.
11. S. Mirjalili, S. M. Mirjalili, and A. Lewis, *Grey Wolf Optimizer*, Advances in Engineering Software, vol. 69, pp. 46–61, 2014.
12. M. Dehghani, E. Trojovská and P. Trojovský, *a new human-based metaheuristic algorithm for solving optimization problems on the base of simulation of driving training process*, Scientific Reports, 2022.
13. M. Dehghani, E. Trojovská & T. Zuščák, *a new human-inspired metaheuristic algorithm for solving optimization problems based on mimicking sewing training*, Scientific Reports, 2022.
14. E. Rashedi, H. Nezamabadi & S. Saryazdi, *a gravitational search algorithm*, Inf. Sci. vol. 179, pp. 2232–2248, 2009.
15. S. Mirjalili, S. M. Mirjalili & Hatamlou, *a multi-verse optimizer : a nature-inspired algorithm for global optimization*, Neural Comput. Appl. vol. 27, pp 495–513, 2016.
16. L. Abualigaha, A. Diabatb, S. Mirjalilid, M. A. Elazizf, A. H. Gandom, *The Arithmetic Optimization Algorithm*, Computer Methods in Applied Mechanics and Engineering, vol. 376, 2021.
17. M. Dehghani, S. Hubálovský and P. Trojovsky, *a new optimization algorithm based on average and subtraction of the best and worst members of the population for solving various optimization problems*, PeerJ Comput. Sci. pp. 2–29, 2022.
18. S. Mirjalili, *A Sine Cosine Algorithm for solving optimization problems*, Knowledge-Based Systems, vol. 00, pp. 1–14, 2016.
19. Van den Bergh F, Engelbrecht A, *A study of particle swarm optimization particle trajectories*, Inf Sci, vol. 176, pp 937–971, 2006.