International Journal of Electrical and Computer Engineering (IJECE)
Vol. 13, No. 1, February 2023, pp. 781~795
ISSN: 2088-8708, DOI: 10.11591/ijece.v13i1.pp781-795

A hybrid swarm intelligence feature selection approach based on time-varying transition parameter

Jomana Yousef Khaseeb[1], Arabi Keshk[2], Anas Youssef[2]
[1]Applied Computing Department, Palestine Technical University/Kadoorie, Tulkarm, Palestine
[2]Faculty of Computers and Information, Menoufia University, Al Menoufia, Egypt

Article Info

Article history:

Received Dec 30, 2021
Revised Jul 18, 2022
Accepted Aug 15, 2022

Keywords:

Grey wolf optimization
Particle swarm optimization
Time varying transition parameter
Tournament selection

ABSTRACT

Feature selection aims to reduce the dimensionality of a dataset by removing superfluous attributes. This paper proposes a hybrid approach for feature selection problem by combining particle swarm optimization (PSO), grey wolf optimization (GWO), and tournament selection (TS) mechanism. Particle swarm enhances the diversification at the beginning of the search mechanism, grey wolf enhances the intensification at the end of the search mechanism, while tournament selection maintains diversification not only at the beginning but also at the end of the search process to achieve local optima avoidance. A time-varying transition parameter and a random variable are used to select either particle swarm, grey wolf, or tournament selection techniques during search process. This paper proposes different variants of this approach based on S-shaped and V-shaped transfer functions (TFs) to convert continuous solutions to binaries. These variants are named hybrid tournament grey wolf particle swarm (HTGWPS), followed by S or V letter to indicate the TF type, and followed by the TF's number. These variants were evaluated using nine high-dimensional datasets. The results revealed that HTGWPS-V1 outperformed other V's variants, PSO, and GWO on 78% of the datasets based on maximum classification accuracy obtained by a minimal feature subset. Also, HTGWPS-V1 outperformed six well-known-metaheuristics on 67% of the datasets.

Corresponding Author:

Jomana Yousef Khaseeb
Applied Computing Department, Palestine Technical University/Kadoorie
Tulkarm, Palestine
Email: Jumanah.khaseeb@ptuk.edu.ps

1. INTRODUCTION

Feature selection (FS) is an essential technique that has been widely utilized to improve the performance of machine learning algorithms in a variety of domains [1]. With the emergence of high-dimensional datasets, FS becomes a difficult challenge because it seeks to find the optimum combination of features that represents the whole set without information loss. According to Liu and Motoda [2] the FS issue utilizes comprehensive search, random search, or heuristic search as search techniques. Comprehensive search produces and examines all probable feature subsets to pick the ideal one, while random search creates and examines stochastic subsets of features. Heuristic search generates a random subset of features and then uses the best solution to guide the search until finding the best feature set. A filter or wrapper approaches are often used to examine the generated subset of features [3]–[5]. The filter method depends on the associations between the features themselves, rather than using any feedback from the learning algorithm [4]. The wrapper model, on the other hand, integrates a learning algorithm as the assessment criteria [6]–[8]. The hybrid method combines both filter and wrapper to take the advantages of both techniques [9], [10].

The literature mentioned a large number of meta-heuristics that are used for optimization issues, such as arithmetic optimization algorithm (AOA) [11], aquila optimizer (AO) [12], reptile search algorithm (RSA) [13], Archimedes optimization algorithm (AOA) [14], archerfish hunting optimizer (AHO) [15], honey badger algorithm (HBA) [16], golden eagle optimizer (GEO) [17], artificial lizard search optimization (ALSO) [18], Harris hawks optimization (HHO) [19], genetic algorithm (GA) [20], particle swarm optimization (PSO) [21], grasshopper optimization algorithm (GOA) [22], bat algorithm (BA) [23], grey wolf optimization (GWO) [24], whale optimization algorithm (WOA) [25], teaching learning based optimization (TLBO) [7], and gravitational search algorithm (GSA) [26].

Exploration (diversification) and exploitation (intensification) are the two objectives that must be balanced in meta-heuristic algorithms to reach the global optima. One method for improving the search methodology and maintaining a balance between exploration and exploitation is by employing hybrid meta-heuristic techniques. Many hybrids meta-heuristic techniques have been proposed in the literature for FS. But, due to the enormous challenges such as high dimensionality, this is still considered an open research subject. An example of a hybrid approach, merging the harmony search (HS) algorithm and the artificial electric field algorithm (AEFA) for FS, which was named the electrical harmony based hybrid meta-heuristic (EHHM) [27]. The obtained findings demonstrated that the EHHM outperformed other state-of-the-art FS techniques as well as their parent algorithms in terms of picking the fewest features with the maximum classification accuracy. Another hybrid algorithm that combines the mayfly algorithm (MA) with HS is presented in [28]. This hybrid algorithm was called mayfly-harmony search (MA-HS) and was utilized for the FS task, attempting to balance the local and global searches to obtain the optimal feature subset. The findings proved that MA-HS obtained high classification accuracy with a low number of features in comparison to the well-known methods from the literature. In study [29], the HHO algorithm was merged with simulated annealing (SA) and chaotic initialization to detect coronavirus disease (COVID-19) from computerized tomography (CT) scan images. In [30], The HPSO-SSM is a hybrid approach that merges PSO with a spiral-shaped mechanism (SSM). This hybrid approach took advantage of the local search ability of the SSM and the global search ability of the PSO to achieve the best feature subset with high classification accuracy. A hybridization of PSO and GWO was proposed in [31]. This approach was called improved grey wolf optimization (IGWO). It utilized the exploratory capability of the PSO to enhance the search mechanism of the GWO to obtain the most relevant features with the minimum classification error without getting stuck in a local optimum.

FS is classified as an optimization problem since it can be used to pick the optimal features from any dataset. Moreover, FS can also be handled as a binary problem since the features can converted to a string of 0's and 1's by using any kind of transfer function (TF) as mentioned in [32], [33]. The meta-heuristic methods may not yield optimal outcomes for all types of problems. This fact is addressed in the No-Free-Lunch theorem [34], which makes this topic an interesting and ongoing research topic for attempting to create an effective optimization strategy. This paper introduces a hybrid meta-heuristic approach that attempts to tune exploration and exploitation to avoid local optima by utilizing PSO, GWO, and tournament selection (TS) [35] as a wrapper-based-FS method. This approach is named the hybrid tournament grey wolf particle swarm algorithm and is abbreviated as (HTGWPS). The main contributions of this hybrid work are as follows: i) PSO is employed in the first steps of the search mechanism helps the algorithm explore the search space for the most promising area; ii) GWO is employed at the end of the search process helps the algorithm exploit the global optima; iii) TS mechanism is employed at the end of the search process in competition with GWO to maintain exploration not only at the beginning but also at the end of the search process to prevent the search algorithm from being trapped in any local optimum; iv) the proposed approach used a time-varying transition parameter and a random variable to orient the search process toward applying PSO, GWO, or TS; and v) the proposed approach was converted into eight binary variants by using four S-shaped and four V-shaped TFs.

The proposed approach was evaluated using nine high-dimensional, low-sample medical datasets [6], [36], [37]. The k-nearest neighbors (KNN) [38] classifier was used for evaluation purposes. The performance of the proposed variants was evaluated against each other, against the original PSO and GWO, and against some well-known FS meta-heuristics from the literature using a set of evaluation metrics such as the average classification accuracy, average selected features, average fitness values, and average computational time. The proposed variant of HTGWPS based on V1 TF outperformed other V-shaped variants, the original PSO, and the original GWO on 78% of the datasets. Moreover, the proposed V1 variant of HTGWPS exceeded six well-known metaheuristics used in the literature for FS on 67% of the datasets in attaining the maximum classification performance acquired by picking the smallest set of features.

The rest of this paper is organized as follows: section 2 introduces the background of the PSO, GWO, TS, and the binary version of the optimization problem. Section 3 describes the details of the proposed approach. Section 4 describes the dataset and the experimental setup. Section 5 presents the experimental results and discussion. Finally, the conclusion and a future direction are summarized in section 6.

2. BACKGROUND
2.1. Particle swarm optimization

PSO was initiated by Kennedy and Eberhart in [39]. It simulates the particles' behavior within the swarm when they are searching for food. Each particle has its own position that refers to a feasible solution in the search space, and its own velocity that manages the particle's movement step and direction during the search process. Each particle is oriented through its movements by the best particle within the swarm and its best location during the search process. The new velocity and the new position of each particle are calculated using (1) and (2) as:

$$vel_n = iw \times vel_o + af_1 \times d_1 \times (per_b - x_{old}) + af_2 \times d_2 \times (glbl_b - x_{old}) \tag{1}$$

$$x_n = vel_n + x_o \tag{2}$$

where vel_n and vel_o represent a particle's new and old velocity, respectively. The per_b and $glbl_b$ are the personal and the global best locations in the search space. The values d_1 and d_2 are two random numbers in the range [0, 1], af_1 and af_2 are acceleration factors that equal to 2 in many studies in the literature. The *iw* is an inertia weight which reduces linearly or non-linearly over the search strategy to optimize exploration and exploitation phases. A new position, x_n, is calculated based on the particle's new velocity, vel_n, and the particle's old position, x_o.

2.2. Grey wolf optimization

The grey wolf optimization (GWO) algorithm was proposed by Mirjalili *et al.* [40]. It mimics the hunting behavior of grey wolves. The GWO initializes the algorithm by generating random positions for each wolf in the population. Each wolf represents a solution in the search area. The evaluation of these solutions is done, and the three best solutions are picked. The best solutions, which are alpha α, beta β, and delta δ, have the better knowledge about the location of the prey. The rest of the pack is called omega. At each iteration, omega wolves follow the three best solutions and changing their locations within the search space to stay close and encircle the best solutions. The mathematical formulas that are used to model these behaviors are as (3)-(5):

$$D_\alpha = |C_1 . X_\alpha - X_w| , D_\beta = |C_2 . X_\beta - X_w|, D_\delta = |C_3 . X_\delta - X_w| \tag{3}$$

$$X_1 = X_\alpha - A_1 . D_\alpha, X_2 = X_\beta - A_2 . D_\beta, X_3 = X_\delta - A_3 . D_\delta \tag{4}$$

$$X_{t+1} = \frac{X_1 + X_2 + X_3}{3} \tag{5}$$

where X_w is wolf's location in the search space. $X_\alpha, X_\beta,$ and X_δ are the positions of the three best solutions. $D_\alpha, D_\beta,$ and D_δ are the distances between each wolf and the best three solutions. X_{t+1} is the new location of each wolf which will be in any random position around the prey. The two coefficients *A* and *C* are calculated by using (6), (7).

$$A = 2a . d_1 - a \tag{6}$$

$$C = 2 . d_2 \tag{7}$$

where d_1 and d_2 are random numbers in the range [0,1], variable *a* decreases linearly from 2 to 0 during the iterations and can be calculated using the:

$$a = 2(1 - \frac{t}{T}) \tag{8}$$

where *a* is a variable that tunes the local and global search of GWO algorithm, *t* is the current iteration, and *T* is the overall number of iterations.

2.3. Tournament selection strategy

The TS technique is employed to choose a guiding solution rather than picking the optimal solution. In this mechanism, a collection of solutions, known as tournament, is chosen randomly then the best solution in the tournament is chosen as a guiding solution [7], [41]. The selected search agent using tournament method is used to guide other agents. This strategy is used to emphasize searching more areas in the large search space. As shown in (9) and (10) show updating the position mechanism:

$$D = |C \,.\, X_{selected} - X| \tag{9}$$

$$X_{(t+1)} = X_{selected} - A \,.\, D \tag{10}$$

where $X_{selected}$ is the search agent selected by the tournament method, D is the distance between each search agent and the selected tournament agent. The two coefficients A and C are calculated by using (6) and (7).

2.4. The binary optimization problem

FS is a binary issue since all attributes in the dataset may be characterized by a string of zeros and ones. Where 0 indicates the unselected features and 1 reflects those that have been picked. According to Mirjalili and Lewis [32], the switch from continuous solutions to binaries is based on two types of TFs, namely S-shaped and V-shaped. The mathematical formulas of these TFs are shown in Table 1.

Table 1. S-shaped and V-shaped TFs [32], [33], [42]

	S-shaped TFs
S1	$T(x) = 1/(1 + e^{-2x})$
S2	$T(x) = 1/(1 + e^{-x})$
S3	$T(x) = 1/(1 + e^{\frac{-x}{2}})$
S4	$T(x) = 1/(1 + e^{\frac{-x}{3}})$
	V-shaped TFs
V1	$T(x) = \left\|\mathrm{erf}\,(\frac{\sqrt{\pi}}{2}\,x)\right\|$
V2	$T(x) = \|\tanh(x)\|$
V3	$T(x) = \left\|\frac{x}{\sqrt{1+x^2}}\right\|$
V4	$T(x) = \left\|\frac{2}{\pi}\arctan(\frac{\pi}{2}x)\right\|$

3. METHOD

The FS issue is a type of optimization problem that looks for the most relevant features in a dataset. In this paper, a hybrid wrapper-based meta-heuristic approach is introduced for the FS task. Search techniques must tune the explorative and exploitative phases to determine the ideal solution. Throughout this paper, PSO with strong exploration potential and GWO with high exploitation capabilities are used, while a TS operator is employed to boost the diversity and the search efficiency to avoid getting the local optima. This approach is named the hybrid tournament grey wolf particle swarm algorithm (HTGWPS). In this paper, the FS issue is treated as a binary problem, which means converting the features into a string of ones and zeros with the same number of attributes as there are in the dataset. A one represents a selected feature, while a zero represents unselected one. In this work, four S-shaped and four V-shaped TFs were used to derive eight variants of the proposed approach. Each variant is named HTGWPS, followed by the type and the number of the TF that was used.

According to this hybrid approach, the initial population is generated randomly. Then, the solutions will be optimized by using the PSO, GWO, or TS techniques. This is examined by using a time-varying transition parameter, namely Z, that is calculated using (11).

$$Z(t) = 1 - \frac{t}{T} \tag{11}$$

where t is the current iteration, and T is the maximum number of iterations.

Figure 1 shows a flowchart of the proposed HTGWPS approach. As shown in the flowchart, the solutions are initialized randomly, then the evaluation of these solutions is performed according to the tackled problem. The parameter Z is calculated. If Z is greater than or equal to 0.5, then the solutions will be updated using PSO mechanism. Otherwise, the solutions will be updated using either GWO or TS depending on a random value, *rand*, between 0 and 1. These steps are repeated until the maximum iteration is reached and the optimal solution is retrieved.

The PSO in the first half of the iterations assists the algorithm in doing a comprehensive scan of the search area by using the PSO's diversification ability to choose the promising area that may hold the global best solution. The GWO technique has intensification properties since it follows the three best solutions throughout the search phase. As a result, the algorithm may get stuck in a local optimum, especially if the algorithm searches a large search space. To address this issue, this paper employs the TS technique with its stochastic behavior in the second half of iterations in competition with the GWO to attain the exploration phase, which prevents this approach from being stuck in a local optimum. Referring to the achieved results, this tactic efficiently tunes between exploration and exploitation and prevents getting trapped in local optima.

Algorithm 1 displays the pseudo-code of the HTGWPS. The lines between 2 and 5 initialize the number of iterations, the population size, and tune the algorithm's parameters. In line 8, Z has been calculated. The lines between 9 and 20, if the value of Z is greater than or equal to 0.5, then the solutions will be improved using the PSO technique based on (1) and (2). Otherwise, a random value, *rand*, has been generated to guide the solutions to update their positions based on the GWO technique using (3) to (8), or based on TS tactic using (9) and (10). These steps are repeated until the maximum number of iterations is reached and the optimal solution is retrieved.

Algorithm 1. The pseudocode of HTGWPS approach

```
1: Start
2: Set the maximum iteration T
3: Set the size of population P
4: Initialize random position and velocity for each individual
5: Initialize the parameters a, A, C, c1, c2
6: t ← 1
7: while t < T do
8:              Calculate Z based on Eq. (11)
9:              p ← 1
10:          for p < P do
11:                  if Z ≥ 0.5 then
12:                            Update the positions according to PSO
13:                 else
14:                      if rand ≤ 0.5 then
15:                                   Update the positions according to GWO
16:                     else
17:                                 Apply TS strategy
18:                     end if
19:                 end if
20:          end for
21:           Save the fittest solutions for the next iteration
22:          t++
23: end while
24: Return the best solution
25: End
```

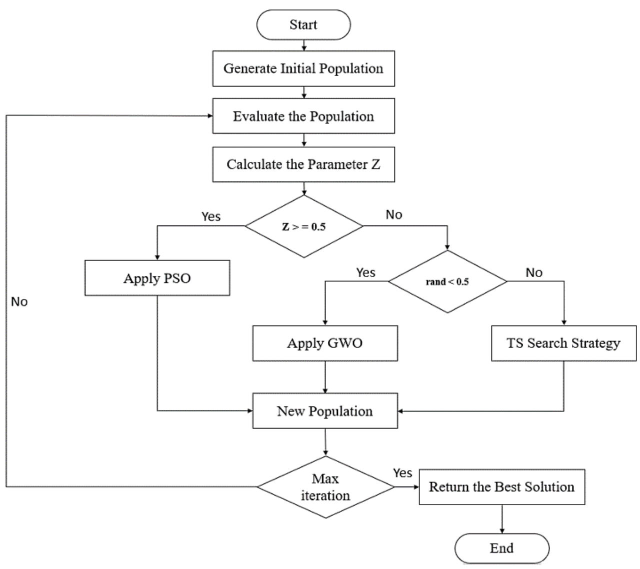

Figure 1. Flowchart of the proposed HTGWPS approach

4. DATASETS AND EXPERIMENTAL SETUP

The approach that is proposed in this paper was evaluated by using nine high-dimensional small-sample medical datasets that are illustrated in Table 2 and mentioned in [6], [36], [37]. Interacting with this form of the dataset is difficult due to the limited number of samples (observations), which makes the training of the learning model inadequate, as well as the huge number of attributes that increase the complexity of the search process in the search scope. The experimental results compared the proposed approach with other investigated approaches in terms of a set of evaluation metrics such as the average number of selected features, the average classification accuracy, the average fitness value, and the average computational time.

Table 2. High-dimensional small-sample medical sets of data

Data set	Instances	Features	Classes
14_Tumors	308	15009	26
11_Tumors	174	12533	11
Brain_Tumor2	50	10367	4
Brain_Tumor1	90	5920	5
Leukemia2	72	11225	3
Leukemia1	72	5327	3
Prostate Tumor	102	10509	2
DLBCL	77	5469	2
SRBCT	83	2308	4

The average classification accuracy is the metric that evaluates the classifier's predictive accuracy using the subset of selected features. It is calculated using (12).

$$Average\ Classification\ Accuracy = \frac{1}{M}\sum_{i=1}^{M}\frac{1}{T}\sum_{j=1}^{T}(P == A) \quad (12)$$

where M is the number of runs, T is the number of instances, A and P are the actual and predictive classes, respectively.

The average selected features are a metric that shows the algorithm's performance in terms of features number while solving the FS problem. This metric is calculated using (13).

$$Average\ Selected\ Features = \frac{1}{M}\sum_{i=1}^{M}\frac{c}{C} \quad (13)$$

Where:
M : the number of runs,
c : the number of selected features, and
C : the overall features.

The average fitness value is the metric that is a combination of the classification error rate and the rate of feature reduction. It is calculated using (14).

$$Average\ Fitness\ Value = \frac{1}{M}\sum_{i=1}^{M}Fit \quad (14)$$

where M is the number of runs, Fit is the fitness value of the best solution in each run, which is calculating using (16).

The average computational time is the amount of time needed to accomplish a computational process. It is calculated using (15).

$$Average\ Computational\ Time = \frac{1}{M}\sum_{i=1}^{M}ComTime \quad (15)$$

where M is the number of runs, $ComTime$ is the computational time.

The quality of a solution is defined by two factors: the number of selected features and the classification error that is recorded by using these features. As shown in (16) shows the objective function that combines these two factors.

$$Fit = m\ cErr + n\ \frac{|f|}{|F|} \quad (16)$$

where *cErr* is the classification error. $|f|$ is the number of the picked attributes, and $|F|$ is the overall attributes in the dataset. The parameter $m \in [0, 1]$ reflects the importance of the classification accuracy, while $n = 1 - m$, which reflects the importance of the feature subset length.

The instances were split randomly into 80% training and 20% testing subsets. The implementation of the proposed approach was done by using MATLAB. The testing was done on a machine with 2.2 GHz Intel Core i7 and 8 GB RAM. The findings were gathered after 30 runs with 100 iterations and 10 search agents. The parameters *m* and *n* in the fitness equation were equal to 0.99 and 0.01, respectively.

5. RESULTS AND DISCUSSION

A set of comparisons were conducted to evaluate the proposed hybrid HTGWPS approach that is proposed in this paper. In section 5.1, the HTGWPS was initially evaluated by comparing the four S-shaped TFs to the original binary versions of PSO and GWO. In section 5.2, the comparisons were conducted to assess the HTGWPS using the four V-shaped TFs with the PSO and GWO parent algorithms. Finally, in section 5.3, the top HTGWPS variants derived from S or V-shaped TFs were compared to well-known metaheuristics from the literature.

5.1. Evaluating HTGWPS variants based on S-shaped TFs

The results of the experimental evaluations of the HTGWPS variants based on S-shaped TFs are described and analyzed in this section. Table 3 displays the classification accuracy of the HTGWPS-S variants. According to the results, HTGWPS-S1 was ranked the best in most datasets. It can be seen that HTGWPS-S1 attained the best classification accuracy in five datasets, and it reached 100% accuracy for Leukemia1 and Leukemia2.

Table 3. The average classification accuracy based on the S-shaped TFs

Dataset	GWO	PSO	HTGWPS-S1	HTGWPS-S2	HTGWPS-S3	HTGWPS-S4
11_Tumors	0.7669	0.7505	**0.7914**	0.7752	0.7752	0.7214
14_Tumors	0.4001	0.4593	0.4410	0.5250	**0.5251**	0.3556
Brain_Tumor1	**0.9444**	0.7611	0.9412	0.7059	0.7059	0.6667
Brain_Tumor2	0.5700	**1.0000**	0.8033	0.9000	0.9000	0.7778
DLBCL	0.8521	0.7958	0.9375	0.8167	0.8167	**1.0000**
Leukemia1	0.9333	0.9267	**1.0000**	0.8622	0.8578	0.9333
Leukemia2	**1.0000**	0.8667	**1.0000**	0.9511	0.9489	**1.0000**
Prostate Tumor	0.9048	0.8571	**0.9841**	0.9524	0.9524	0.9048
SRBCT	0.9059	0.9412	**0.9431**	0.8588	0.8588	0.8020
Mean Rank	3.44	4.00	**2.00**	3.56	3.67	4.33
Rank	2	5	**1**	3	4	6

Table 4 shows the average number of features chosen by each variant for each dataset. The table shows that HTGWPS-S2 outperformed the other competitors in five datasets and recorded the best mean rank, followed by HTGWPS-S4. According to Table 5, the average fitness results also indicate that HTGWPS-S1 achieved the best approach based on S-shaped TF by choosing the minimum relevant attributes with the lowest classification error. The outcomes revealed that the HTGWPS-S1 is the fittest technique based on the S-shaped TFs, and this is evidenced by the value of the mean rank. Table 6 shows that PSO is the best approach in terms of the average computational time, followed by GWO, and this superiority is due to the hybridization of the proposed approach, which needs more computational time.

Table 4. The average number of selected features based on the S-shaped TFs

Dataset	GWO	PSO	HTGWPS-S1	HTGWPS-S2	HTGWPS-S3	HTGWPS-S4
11_Tumors	7879.90	6182.17	6315.30	6067.30	6064.80	**5998.60**
14_Tumors	9752.07	**7475.73**	8364.57	7720.10	7640.30	7854.57
Brain_Tumor1	2897.17	2810.20	2650.30	2457.70	2492.87	**2380.20**
Brain_Tumor2	5859.93	4947.93	4907.20	**4427.60**	4496.93	4515.00
DLBCL	3036.77	2611.17	2535.43	**1680.63**	2476.17	2390.63
Leukemia1	2619.57	2571.10	2534.03	**1878.93**	2559.40	2458.90
Leukemia2	6771.23	5349.20	5041.90	4796.97	5203.53	**4688.60**
Prostate_Tumor	5468.77	4995.07	5159.07	**4522.03**	4877.47	4929.53
SRBCT	1422.07	1094.27	1029.83	**996.33**	1070.07	1181.87
Mean Rank	6.00	4.22	3.89	**1.67**	2.78	2.44
Rank	6	5	4	**1**	3	2

Table 5. The average fitness values based on the S-shaped TFs

Dataset	GWO	PSO	HTGWPS-S1	HTGWPS-S2	HTGWPS-S3	HTGWPS-S4
11_Tumors	0.2370	0.2520	**0.2113**	0.2273	0.2274	0.2806
14_Tumors	0.6004	0.5403	0.5586	0.4753	**0.4752**	0.6432
Brain_Tumor1	**0.0599**	0.2413	0.0624	0.2954	0.2954	0.3340
Brain_Tumor2	0.4314	**0.0048**	0.1990	0.1034	0.1033	0.2244
DLBCL	0.1520	0.2069	0.0649	0.1860	0.1860	**0.0044**
Leukemia1	0.0709	0.0774	**0.0035**	0.1412	0.1456	0.0706
Leukemia2	0.0060	0.1368	0.0043	0.0531	0.0552	**0.0042**
Prostate_Tumor	0.0995	0.1462	**0.0200**	0.0519	0.0518	0.0990
SRBCT	0.0993	0.0630	**0.0606**	0.1445	0.1444	0.2012
Mean Rank	3.67	4.00	**2.00**	3.67	3.56	4.11
Rank	3	4	**1**	3	2	5

Table 6. The average computational time based on S-shaped TFs

Dataset	GWO	PSO	HTGWPS-S1	HTGWPS-S2	HTGWPS-S3	HTGWPS-S4
11_Tumors	334.54	**259.24**	519.50	353.96	672.48	408.50
14_Tumors	1125.54	871.78	1838.33	2163.07	2039.68	**808.25**
Brain_Tumor1	38.35	**33.32**	127.81	175.55	136.38	71.61
Brain_Tumor2	37.04	**32.55**	180.72	123.52	103.18	126.86
DLBCL	32.02	**28.01**	106.45	100.94	66.50	64.75
Leukemia1	30.06	**26.64**	96.03	61.77	56.50	57.55
Leukemia2	57.27	**45.74**	225.40	127.35	111.75	149.20
Prostate_Tumor	96.79	**65.77**	276.48	150.18	116.92	155.96
SRBCT	20.21	**18.86**	48.77	66.42	26.53	26.15
Mean Rank	2.11	**1.11**	5.33	4.78	4.00	3.67
Rank	2	**1**	6	5	4	3

In this paper, the accuracy rate has a higher priority than the number of selected attributes. So, according to the outcomes, the HTGWPS based on S1-TF yielded better performance. The findings revealed that HTGWPS-S1 offers a stable trade-off between exploration and exploitation to escape local optima and attain higher fitness results. The power of the proposed approach is the high exploration ability at the beginning of the search process by using PSO optimizers, while utilizing GWO gives the approach the exploitation ability after passing half the number of iterations. The TS tactic is utilized to give the exploration a chance at the end of the search process to avoid being stuck in the local optima.

5.2. Evaluating HTGWPS variants based on V-shaped TFs

This section presents the results of the HTGWPS variants based on V-shaped TFs in terms of the four-evaluation metrics. According to Table 7, it can be seen that HTGWPS-V1 recorded the best average classification accuracy on 89% of the dataset, while HTGWPS-V4 was rated second by the mean rank value. Table 8 displays the average number of features picked by each V variant. According to the results, GWO chose the fewest features in five datasets, followed by HTGWPS-V1 in four datasets. However, when the mean rank was determined, both techniques had the same overall rank across the nine datasets included in this paper. Table 9 summarizes the average fitness values of all V-shaped HTGWPS techniques. As mentioned previously, the fitness values represent the lowest classification error that is obtained from the least selected features. Whereas the strategy with the smallest feature subset size and the lowest classification error is the most effective. Here, we can see that HTGWPS-V1 was rated first in 78% of the datasets. This indicates that the HTGWPS-V1 optimizer can perform a consistent trade-off between exploration and exploitation in order to reach the global optima.

Table 7. The average classification accuracy based on V-shaped TFs

Dataset	GWO	PSO	HTGWPS-V1	HTGWPS-V2	HTGWPS-V3	HTGWPS-V4
11_Tumors	0.8514	0.8400	**0.8874**	0.7876	0.7800	0.8675
14_Tumors	0.5067	0.5294	**0.6088**	0.5521	0.5521	0.5927
Brain_Tumor1	0.7815	0.8333	**0.8464**	0.7078	0.7118	0.8049
Brain_Tumor2	0.9433	0.8000	**1.0000**	0.9100	0.9033	**1.0000**
DLBCL	0.9896	0.9208	**0.9938**	0.8688	0.8646	0.9917
Leukemia1	0.9067	0.9333	**0.9867**	0.8622	0.8622	0.9533
Leukemia2	**1.0000**	0.8733	**1.0000**	0.9733	0.9911	**1.0000**
Prostate_Tumor	**1.0000**	0.8524	**1.0000**	0.9540	0.9540	0.9968
SRBCT	**1.0000**	0.9294	0.9961	0.8961	0.8941	0.9941
Mean Rank	3.06	4.44	**1.33**	4.83	5.06	2.82
Rank	3	4	**1**	5	6	2

Table 8. The average number of selected features based on V-shaped TFs

Dataset	GWO	PSO	HTGWPS-V1	HTGWPS-V2	HTGWPS-V3	HTGWPS-V4
11_Tumors	646.10	6029.03	**342.10**	3871.13	3923.03	533.03
14_Tumors	**785.13**	7413.20	1655.07	5720.07	5868.63	1881.17
Brain_Tumor1	**7.93**	2710.47	15.33	937.33	976.93	38.23
Brain_Tumor2	90.57	4971.47	**69.03**	2144.60	2265.73	148.57
DLBCL	68.10	2564.53	**66.73**	1333.70	1373.27	145.03
Leukemia1	99.63	2476.23	**91.00**	1531.47	1536.50	163.63
Leukemia2	**132.23**	5281.67	199.20	2994.13	3114.70	350.07
Prostate_Tumor	**88.13**	5004.73	103.30	2582.43	2799.80	210.83
SRBCT	**49.17**	1046.87	50.20	618.80	642.83	81.77
Mean Rank	**1.56**	6.00	**1.56**	4.00	5.00	2.89
Rank	**1**	5	**1**	3	4	2

Table 9. The average fitness values based on V-shaped TFs

Dataset	GWO	PSO	HTGWPS-V1	HTGWPS-V2	HTGWPS-V3	HTGWPS-V4
11_Tumors	0.1477	0.1632	**0.1118**	0.2134	0.2209	0.1316
14_Tumors	0.4889	0.4708	**0.3884**	0.4472	0.4473	0.4045
Brain_Tumor1	0.2164	0.1696	**0.1521**	0.2908	0.2870	0.1932
Brain_Tumor2	0.0562	0.2028	**0.0001**	0.0912	0.0979	0.0001
DLBCL	0.0104	0.0831	**0.0063**	0.1324	0.1366	0.0085
Leukemia1	0.0926	0.0706	**0.0134**	0.1393	0.1393	0.0465
Leukemia2	**0.0001**	0.1301	0.0002	0.0291	0.0116	0.0003
Prostate_Tumor	**0.0001**	0.1509	**0.0001**	0.0480	0.0482	0.0033
SRBCT	**0.0002**	0.0744	0.0041	0.1056	0.1076	0.0062
Mean Rank	2.94	4.44	**1.33**	4.72	5.17	2.39
Rank	3	4	**1**	5	6	2

Table 10 displays the average computational times. It is demonstrated that the GWO obtained the shortest time for all datasets, followed by the PSO. While the proposed variants need more computational time due to the hybridization technique. Figure 2 shows the convergence curves of the fitness values during 100 iterations for HTGWPS-V1, GWO, and PSO in dealing with different datasets that are used in this paper. According to these curves, the HTGWPS-V1 obtained the best results on 7 out of 9 datasets. This is evident in Figures 2(a) to (i), since HTGWPS-V1 achieved the lowest fitness values in most curves.

Figure 2(a) describes the behavior of the three algorithms on the 11-Tumors dataset. It shows that HTGWPS-V1 curves down according to the iterations until it reaches the lowest fitness value, followed by GWO, while PSO stuck in a local optimum. Figure 2(b) shows the fitness value curves of three approaches on the 14-Tumors dataset. It also shows that HTGWPS-V1 optimizes the results by minimizing the number of features with a minimum classification error during the iterations, followed by PSO, which drops at a local minimum value.

Figure 2(c) shows the convergence curves of the fitness values of three optimizer approaches on the Brain-Tumor1 dataset. The curves show that HTGWPS-V1 obtained the minimum fitness value after scanning the search space, and this is obvious by the behavior of the curve, which slopes down during the iterations while PSO is stuck in the local optima in early iterations, and GWO had the highest fitness values. Figure 2(d) shows the fitness value curves of three approaches on the Brain-Tumor2 dataset. It shows that HTGWPS-V1 achieved the best fitness value after 40 iterations. As seen in the HTGWPS-V1 curve, the values of fitness values are near to zero, which means the algorithm reaches the minimum number of features with a minimum classification error.

Table 10. The average computational time based on V-shaped TFs

Dataset	GWO	PSO	HTGWPS-V1	HTGWPS-V2	HTGWPS-V3	HTGWPS-V4
11_Tumors	**53.75**	550.25	180.33	277.76	270.18	309.76
14_Tumors	**155.77**	1353.26	533.38	1448.22	1834.33	1066.61
Brain_Tumor1	**18.14**	45.10	126.70	76.83	108.18	185.10
Brain_Tumor2	**19.66**	51.32	197.83	64.02	134.49	188.24
DLBCL	**17.38**	39.62	304.20	34.23	81.01	86.33
Leukemia1	**17.17**	44.88	129.57	41.38	48.41	93.15
Leukemia2	**22.28**	71.70	187.03	78.78	85.93	258.73
Prostate_Tumor	**23.74**	64.60	142.81	95.42	99.99	225.89
SRBCT	**15.18**	19.00	44.06	29.58	28.34	35.60
Mean Rank	**1.00**	2.89	4.78	3.22	4.00	5.11
Rank	**1**	2	5	3	4	6

Figure 2(e) shows the behavior of the HTGWPS-V1, GWO, and PSO on the DLBCL dataset. The curves show that HTGWPS-V1 can reach the minimum fitness value, followed by GWO, then PSO. Figure 2(f) illustrates the behavior of the three algorithms on the Leukemia1 dataset. HTGWPS-V1 obtained the minimum fitness value during the iterations, followed by PSO, which stuck in local optima after 20 iterations, followed by GWO. Figure 2(g) shows the fitness value curves when dealing with the Leukemia2 dataset. The figure shows that GWO and HTGWPS-V1 have the minimum number of features with maximum classification accuracy, while PSO cannot clearly optimize its result during 100 iterations. Figure 2(h) describes the behavior of the three optimizers on the Prostate-Tumor dataset. The curves show that HTGWPS-V1 and GWO obtained the minimum fitness values, which is near to zero, while PSO cannot reach the best results during 100 iterations and trapped in the local optima. Figure 2(i) shows the fitness value curves of the three optimizers in the SRBCT dataset. It also shows the best results obtained by GWO and HTGWPS-V1 with few differences, followed by PSO optimizer.

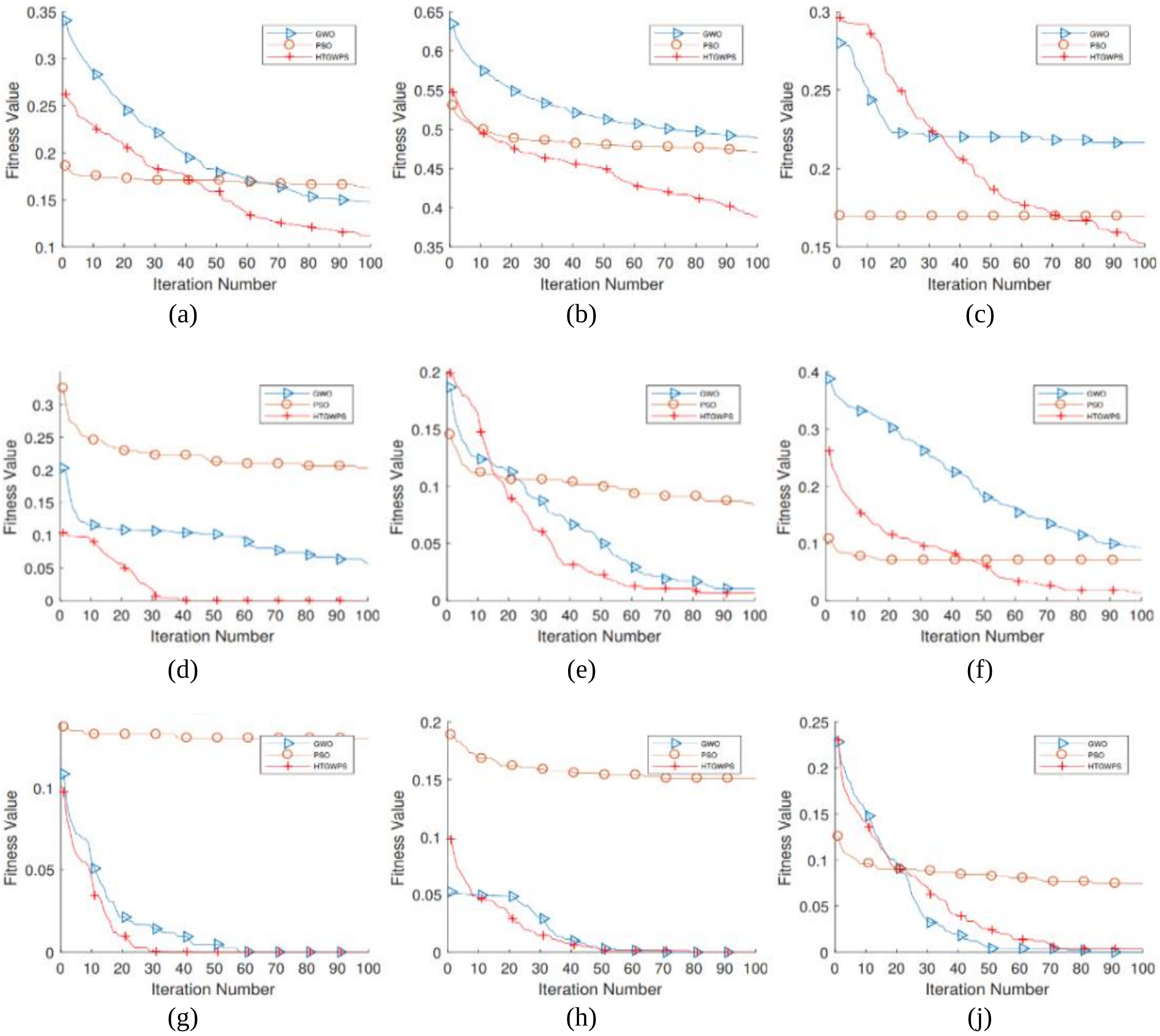

Figure 2. Convergence curves of GWO, PSO, and HTGWPS-V1 using V-Shaped TF on different datasets: (a) 11-Tumors, (b) 14-Tumors, (c) Brain-Tumor1, (d) Brain-Tumor2, (e) DLBCL, (f) Leukemia1, (g) Leukemia2, (h) Prostate-Tumor, and (i) SRBCT

5.3. Comparing top HTGWPS variants with metaheuristics from the literature

In this section, a comparison of the top performing variants in both S-shaped and V-shaped with well-established optimizers from the literature such as GSA, ant lion optimization (ALO) [33], BA, WOA, HHO, and TLBO has been carried out. According to the preceding two sections, HTGWPS-S1 had the best performing S-shaped approach, whereas HTGWPS-V1 had the best performing V-shaped approach.

Table 11 clearly shows that HTGWPS-V1 outperformed all other techniques in terms of classification accuracy, as it produced the best results in five datasets, with 100% accuracy on three datasets. The HTGWPS-S1 outperformed other techniques in three datasets. In terms of the average number of features, Table 12 shows that WOA performed the best in seven datasets, followed by HTGWPS-V1 in two datasets, with a little difference between their results. Table 13 reveals that HTGWPS-V1 outperformed other techniques in terms of average fitness values in six datasets, followed by HTGWPS-S1. The results showed that V1 is the best TF that is used to binarize HTGWPS, which revealed the superiority in tuning between global and local search to avoid the local optima and attain the global solution.

Table 11. Comparison of HTGWPS-S1, HTGWPS-V1, and other meta-heuristics in terms of classification accuracy

Dataset	HTGWPS-S1	HTGWPS-V1	GSA	ALO	BA	WOA	HHO	TLBO
11_Tumors	0.7914	0.8874	0.61875	0.79218	0.72997	0.86286	0.88952	0.85810
14_Tumors	0.4410	**0.6088**	0.47718	0.50003	0.39370	0.40985	0.51320	0.53875
Brain_Tumor1	**0.9412**	0.8464	0.77778	0.77778	0.85185	0.77778	0.92593	0.72222
Brain_Tumor2	0.8033	**1.0000**	0.72333	0.88889	0.45000	0.62000	0.90000	0.80000
DLBCL	0.9375	0.9938	**1.00000**	**1.00000**	0.88958	0.93750	**1.00000**	0.87500
Leukemia1	**1.0000**	0.9867	0.86667	0.86000	0.93111	0.80000	0.87111	0.86667
Leukemia2	**1.0000**	**1.0000**	0.87778	**1.00000**	0.74222	0.93333	0.93333	**1.00000**
Prostate_Tumor	0.9841	**1.0000**	0.76984	0.90476	0.84921	0.95238	0.80952	0.87778
SRBCT	0.9431	**0.9961**	0.85882	0.83922	0.80000	0.92941	0.91569	0.89020
Mean Rank	3.33	**2.06**	5.94	4.50	6.44	5.33	3.39	5.00
Rank	2	**1**	7	4	8	6	3	5

Table 12. Comparison between HTGWPS-S1, HTGWPS-V1, and other meta-heuristics in terms of selected features

Dataset	HTGWPS-S1	HTGWPS-V1	GSA	ALO	BA	WOA	HHO	TLBO
11_Tumors	6315.30	342.10	6301.13	8121.93	4924.50	**322.76**	504.77	1036.05
14_Tumors	8364.57	1655.07	7580.40	11818.33	6060.00	**1091.01**	1689.28	2122.66
Brain_Tumor1	2650.30	**15.33**	2894.20	2900.90	2435.80	44.33	65.18	73.25
Brain_Tumor2	4907.20	69.03	5175.13	7752.23	4223.17	**62.08**	95.20	76.88
DLBCL	2535.43	66.73	2665.13	2688.30	2176.03	**37.25**	110.97	63.61
Leukemia1	2534.03	91.00	2628.30	3717.07	2035.77	**34.79**	97.67	61.01
Leukemia2	5041.90	199.20	5558.93	5529.17	4585.97	**82.84**	201.69	99.11
Prostate_Tumor	5159.07	**103.30**	5221.27	5202.23	4108.93	105.78	370.09	131.93
SRBCT	1029.83	50.20	1138.90	1384.60	925.00	**19.80**	64.33	41.20
Mean Rank	6.22	2.22	7.00	7.78	5.00	**1.22**	3.67	2.89
Rank	6	2	7	8	5	**1**	4	3

Table 13. Comparison between HTGWPS-S1, HTGWPS-V1, and other meta-heuristics in terms of fitness values

Dataset	HTGWPS-S1	HTGWPS-V1	GSA	ALO	BA	WOA	HHO	TLBO
11_Tumors	0.2113	**0.1118**	0.3825	0.2122	0.2294	0.1403	0.1137	0.1451
14_Tumors	0.5586	**0.3884**	0.5226	0.5029	0.5771	0.5900	0.4874	0.4614
Brain_Tumor1	**0.0624**	0.1521	0.2249	0.2249	0.1140	0.2231	0.0771	0.2793
Brain_Tumor2	0.1990	**0.0001**	0.2789	0.1175	0.4821	0.3802	0.1022	0.2023
DLBCL	0.0649	0.0063	0.0049	0.0049	0.0739	0.0650	**0.0027**	0.1281
Leukemia1	**0.0035**	0.0134	0.1369	0.1456	0.0693	0.2012	0.1303	0.1363
Leukemia2	0.0043	**0.0002**	0.1260	0.0049	0.2302	0.0695	0.0691	0.0043
Prostate_Tumor	0.0200	**0.0001**	0.2328	0.0992	0.1448	0.0514	0.1916	0.1255
SRBCT	0.0606	**0.0041**	0.1447	0.1652	0.1512	0.0747	0.0869	0.1132
Mean Rank	3.17	**1.78**	6.11	5.00	6.22	5.44	3.33	4.94
Rank	2	**1**	7	5	8	6	3	4

Table 14 displays the average computational times. It demonstrated that the BA obtained the shortest time on 78% of the datasets. The comparisons presented in this section are considered fair comparisons since all techniques were run in the same experimental environment. The comparisons clearly showed that HTGWPS-V1 outperforms other approaches in terms of average fitness value, which combines the size of the features that are picked and the classification error that is obtained from these features.

The binary version of HTGWPS based on V1 TF obtained very competitive results compared to other competitors, followed by HTGWPS based on S1 TF. The main reason is that HTGWPS can achieve a more stable balance between exploration and exploitation. It switches effectively between exploration and exploitation using the time-varying parameter, *Z*, and the random parameter, *rand*, to produce more exploitative behavior not only in the early stages of the search process, but also in the last stages. This mechanism reveals the potential of this approach to jump out of local optima.

Table 14. Comparison of HTGWPS-S1, HTGWPS-V1, and other meta-heuristics in terms of computational time

Dataset	HTGWPS-S1	HTGWPS-V1	GSA	ALO	BA	WOA	HHO	TLBO
11_Tumors	519.50	**180.33**	476.58	424.82	453.79	322.76	504.77	1036.05
14_Tumors	1838.33	**533.38**	901.57	2438.23	795.30	1091.01	1689.28	2122.66
Brain_Tumor1	127.81	126.70	37.35	139.05	**32.07**	44.33	65.18	73.25
Brain_Tumor2	180.72	197.83	40.35	123.11	**30.65**	62.08	95.20	76.88
DLBCL	106.45	304.20	32.51	96.90	**26.28**	37.25	110.97	63.61
Leukemia1	96.03	129.57	30.86	93.30	**24.82**	34.79	97.67	61.01
Leukemia2	225.40	187.03	54.49	143.82	**43.86**	82.84	201.69	99.11
Prostate_Tumor	276.48	142.81	75.01	221.23	**59.39**	105.78	370.09	131.93
SRBCT	48.77	44.06	20.29	33.25	**17.84**	19.80	64.33	41.20
Mean Rank	6.78	5.44	2.56	5.56	**1.44**	2.89	6.33	5.00
Rank	8	5	2	6	**1**	3	7	4

5.4. Impact of feature selection on the classification performance

This section shows a comparison between using the KNN classifier with HTGWPS versus without using this novel approach. Based on Table 15, using HTGWPS-V1 with KNN increases the performance of the classifier in terms of classification accuracy and the number of relevant features selected. This is also clearly observed in Figure 3.

Table 15. Comparison of classification accuracy using KNN and HTGWPS

Dataset	Classification Accuracy		Number of Features	
	KNN ONLY	KNN with HTGWPS-V1	KNN ONLY	KNN with HTGWPS-V1
11_Tumors	0.7072	**0.8874**	15009	**342**
14_Tumors	0.3968	**0.6088**	12533	**1655**
Brain_Tumor1	0.7700	**0.8464**	10367	**15**
Brain_Tumor2	0.6422	**1.0000**	5920	**69**
DLBCL	0.8625	**0.9938**	11225	**67**
Leukemia1	0.8267	**0.9867**	5327	**91**
Leukemia2	0.8667	**1.0000**	10509	**199**
Prostate_Tumor	0.8349	**1.0000**	5469	**103**
SRBCT	0.8587	**0.9961**	2308	**50**

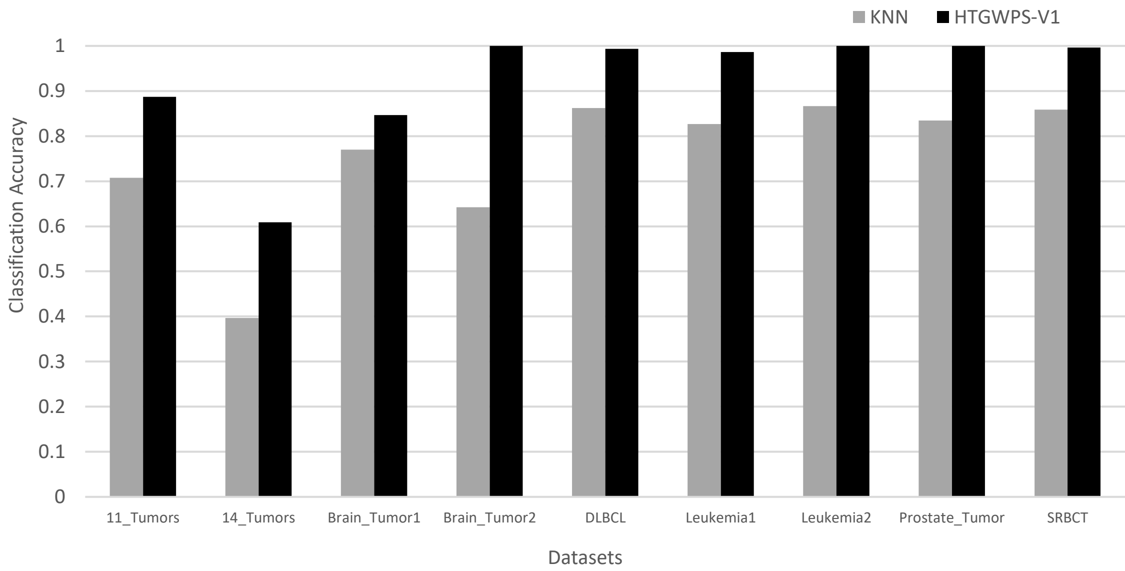

Figure 3. Accuracy rates before and after applying HTGWPS-based FS

6. CONCLUSION

A hybrid approach that is proposed in this paper is used to handle the complexity of the FS problem in large datasets. This hybrid approach combines PSO, GWO, and TS techniques in the proper way to avoid the local optima, especially when dealing with high-dimensional datasets. The PSO's exploration ability at the beginning step of the search mechanism helps the algorithm scan the search space for the promising search area. The exploitation capability of the GWO at the end of the search mechanism helps the algorithm to converge toward the best solution. The TS applies the exploratory phase at the end of the search process to escape from local optima and tend toward the global optima. This approach uses a time-varying transition parameter, named Z. Depending on Z's threshold, the solutions are updated based on the PSO, GWO, or TS. The FS is treated as a binary problem by using the TFs to convert continuous solutions to binaries. Eight variants of the proposed approach were implemented in this paper. Four variants are based on S-shaped TFs and four are based on V-shaped TFs. To evaluate the proposed variants, a variety of high-dimensional small-instance medical datasets were used, and the KNN classifier's feedback was utilized to assess the performance of the proposed approach. The experimental findings demonstrated that HTGWPS-V1 outperformed the other investigated variants and well-known optimizers in terms of average fitness value on 67% of the datasets. Future work includes utilizing the TS mechanism with other meta-heuristics that suffer from global search ability. Also, the Z-parameter can be tuned to other thresholds, which may obtain better results.

ACKNOWLEDGEMENTS

The authors would like to thank Palestine Technical University-Kadoorie for their financial support to conduct this research.

REFERENCES

[1] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of machine learning research*, vol. 3, pp. 1157–1182, 2003.

[2] H. Liu and H. Motoda, *Feature selection for knowledge discovery and data mining*. Boston, MA: Springer US, 1998.

[3] J. Tang, S. Alelyani, and H. Liu, *Feature selection for classification: A review*. Chapman and Hall/CRC, 2014.

[4] L. Brezočnik, I. Fister, and V. Podgorelec, "Swarm intelligence algorithms for feature selection: A review," *Applied Sciences*, vol. 8, no. 9, Sep. 2018, doi: 10.3390/app8091521.

[5] M. Mafarja *et al.*, "Binary dragonfly optimization for feature selection using time-varying transfer functions," *Knowledge-Based Systems*, vol. 161, pp. 185–204, Dec. 2018, doi: 10.1016/j.knosys.2018.08.003.

[6] T. Thaher, A. A. Heidari, M. Mafarja, J. S. Dong, and S. Mirjalili, "Binary Harris hawks optimizer for high-dimensional, low sample size feature selection," in *Evolutionary Machine Learning Techniques, Algorithms for Intelligent Systems*, 2020, pp. 251–272.

[7] T. Thaher, M. Mafarja, H. Turabieh, P. A. Castillo, H. Faris, and I. Aljarah, "Teaching learning-based optimization with evolutionary binarization schemes for tackling feature selection problems," *IEEE Access*, vol. 9, pp. 41082–41103, 2021, doi: 10.1109/ACCESS.2021.3064799.

[8] A. Wang, N. An, G. Chen, L. Li, and G. Alterovitz, "Accelerating wrapper-based feature selection with K-nearest-neighbor," *Knowledge-Based Systems*, vol. 83, pp. 81–91, Jul. 2015, doi: 10.1016/j.knosys.2015.03.009.

[9] H.-H. Hsu, C.-W. Hsieh, and M.-D. Lu, "Hybrid feature selection by combining filters and wrappers," *Expert Systems with Applications*, vol. 38, no. 7, pp. 8144–8150, Jul. 2011, doi: 10.1016/j.eswa.2010.12.156.

[10] G. Naqvi, "A hybrid filter-wrapper approach for feature selection," Master's Thesis, Örebro University, 2012.

[11] L. Abualigah, A. Diabat, S. Mirjalili, M. Abd Elaziz, and A. H. Gandomi, "The arithmetic optimization algorithm," *Computer Methods in Applied Mechanics and Engineering*, vol. 376, Apr. 2021, doi: 10.1016/j.cma.2020.113609.

[12] L. Abualigah, D. Yousri, M. Abd Elaziz, A. A. Ewees, M. A. A. Al-qaness, and A. H. Gandomi, "Aquila optimizer: A novel meta-heuristic optimization algorithm," *Computers & Industrial Engineering*, vol. 157, Art. no. 107250, Jul. 2021, doi: 10.1016/j.cie.2021.107250.

[13] L. Abualigah, M. A. Elaziz, P. Sumari, Z. W. Geem, and A. H. Gandomi, "Reptile search algorithm (RSA): A nature-inspired meta-heuristic optimizer," *Expert Systems with Applications*, vol. 191, Apr. 2022, doi: 10.1016/j.eswa.2021.116158.

[14] 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," *Applied Intelligence*, vol. 51, no. 3, pp. 1531–1551, Mar. 2021, doi: 10.1007/s10489-020-01893-z.

[15] F. Zitouni, S. Harous, A. Belkeram, and L. E. B. Hammou, "The archerfish hunting optimizer: A novel metaheuristic algorithm for global optimization," *Arabian Journal for Science and Engineering*, vol. 47, no. 2, pp. 2513–2553, Feb. 2022, doi: 10.1007/s13369-021-06208-z.

[16] F. A. Hashim, E. H. Houssein, K. Hussain, M. S. Mabrouk, and W. Al-Atabany, "Honey badger algorithm: New metaheuristic algorithm for solving optimization problems," *Mathematics and Computers in Simulation*, vol. 192, pp. 84–110, Feb. 2022, doi: 10.1016/j.matcom.2021.08.013.

[17] A. Mohammadi-Balani, M. Dehghan Nayeri, A. Azar, and M. Taghizadeh-Yazdi, "Golden eagle optimizer: A nature-inspired metaheuristic algorithm," *Computers & Industrial Engineering*, vol. 152, Feb. 2021, doi: 10.1016/j.cie.2020.107050.

[18] N. Kumar, N. Singh, and D. P. Vidyarthi, "Artificial lizard search optimization (ALSO): a novel nature-inspired meta-heuristic algorithm," *Soft Computing*, vol. 25, no. 8, pp. 6179–6201, Apr. 2021, doi: 10.1007/s00500-021-05606-7.

[19] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Future Generation Computer Systems*, vol. 97, pp. 849–872, Aug. 2019, doi: 10.1016/j.future.2019.02.028.

[20] J. Huang, Y. Cai, and X. Xu, "A hybrid genetic algorithm for feature selection wrapper based on mutual information," *Pattern Recognition Letters*, vol. 28, no. 13, pp. 1825–1844, Oct. 2007, doi: 10.1016/j.patrec.2007.05.011.

[21] B. Xue, M. Zhang, and W. N. Browne, "Particle swarm optimization for feature selection in classification: A multi-objective approach," *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 1656–1671, Dec. 2013, doi: 10.1109/TSMCB.2012.2227469.

[22] M. Mafarja *et al.*, "Evolutionary population dynamics and grasshopper optimization approaches for feature selection problems," *Knowledge-Based Systems*, vol. 145, pp. 25–45, Apr. 2018, doi: 10.1016/j.knosys.2017.12.037.

[23] X. S. Yang and X. He, "Bat algorithm: literature review and applications," *International Journal of Bio-Inspired Computation*, vol. 5, no. 3, 2013, doi: 10.1504/IJBIC.2013.055093.

[24] Q. Al-Tashi, S. J. Abdul Kadir, H. M. Rais, S. Mirjalili, and H. Alhussian, "Binary optimization using hybrid grey wolf optimization for feature selection," *IEEE Access*, vol. 7, pp. 39496–39508, 2019, doi: 10.1109/ACCESS.2019.2906757.

[25] M. Mafarja and S. Mirjalili, "Whale optimization approaches for wrapper feature selection," *Applied Soft Computing*, vol. 62, pp. 441–453, Jan. 2018, doi: 10.1016/j.asoc.2017.11.006.

[26] M. Taradeh *et al.*, "An evolutionary gravitational search-based feature selection," *Information Sciences*, vol. 497, pp. 219–239, Sep. 2019, doi: 10.1016/j.ins.2019.05.038.

[27] K. H. Sheikh *et al.*, "EHHM: Electrical harmony based hybrid meta-heuristic for feature selection," *IEEE Access*, vol. 8, pp. 158125–158141, 2020, doi: 10.1109/ACCESS.2020.3019809.

[28] T. Bhattacharyya, B. Chatterjee, P. K. Singh, J. H. Yoon, Z. W. Geem, and R. Sarkar, "Mayfly in harmony: A new hybrid meta-heuristic feature selection algorithm," *IEEE Access*, vol. 8, pp. 195929–195945, 2020, doi: 10.1109/ACCESS.2020.3031718.

[29] R. Bandyopadhyay, A. Basu, E. Cuevas, and R. Sarkar, "Harris Hawks optimisation with simulated annealing as a deep feature selection method for screening of COVID-19 CT-scans," *Applied Soft Computing*, vol. 111, Nov. 2021, doi: 10.1016/j.asoc.2021.107698.

[30] K. Chen, F.-Y. Zhou, and X.-F. Yuan, "Hybrid particle swarm optimization with spiral-shaped mechanism for feature selection," *Expert Systems with Applications*, vol. 128, pp. 140–156, Aug. 2019, doi: 10.1016/j.eswa.2019.03.039.

[31] J. Yousef, A. Youssef, and A. Keshk, "A hybrid swarm intelligence based feature selection algorithm for high dimensional datasets," *IJCI. International Journal of Computers and Information*, vol. 8, no. 1, pp. 67–86, May 2021, doi: 10.21608/ijci.2021.62499.1040.

[32] S. Mirjalili and A. Lewis, "S-shaped versus V-shaped transfer functions for binary particle swarm optimization," *Swarm and Evolutionary Computation*, vol. 9, pp. 1–14, Apr. 2013, doi: 10.1016/j.swevo.2012.09.002.

[33] M. Mafarja, D. Eleyan, S. Abdullah, and S. Mirjalili, "S-shaped vs. V-shaped transfer functions for ant lion optimization algorithm in feature selection problem," in *Proceedings of the International Conference on Future Networks and Distributed Systems*, Jul. 2017, pp. 1–7, doi: 10.1145/3102304.3102325.

[34] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, Apr. 1997, doi: 10.1109/4235.585893.

[35] M. M. Mafarja and S. Mirjalili, "Hybrid whale optimization algorithm with simulated annealing for feature selection," *Neurocomputing*, vol. 260, pp. 302–312, Oct. 2017, doi: 10.1016/j.neucom.2017.04.053.

[36] A. Statnikov, C. F. Aliferis, I. Tsamardinos, D. Hardin, and S. Levy, "A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis," *Bioinformatics*, vol. 21, no. 5, pp. 631–643, Mar. 2005, doi: 10.1093/bioinformatics/bti033.

[37] M. Mafarja, I. Jaber, S. Ahmed, and T. Thaher, "Whale optimisation algorithm for high-dimensional small-instance feature selection," *International Journal of Parallel, Emergent and Distributed Systems*, vol. 36, no. 2, pp. 80–96, Mar. 2021, doi: 10.1080/17445760.2019.1617866.

[38] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, Aug. 1992, doi: 10.1080/00031305.1992.10475879.

[39] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, pp. 1942–1948, doi: 10.1109/ICNN.1995.488968.

[40] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, Mar. 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[41] M. Mafarja, A. A. Heidari, M. Habib, H. Faris, T. Thaher, and I. Aljarah, "Augmented whale feature selection for IoT attacks: Structure, analysis and applications," *Future Generation Computer Systems*, vol. 112, pp. 18–40, Nov. 2020, doi: 10.1016/j.future.2020.05.020.

[42] K. K. Ghosh, R. Guha, S. K. Bera, N. Kumar, and R. Sarkar, "S-shaped versus V-shaped transfer functions for binary Manta ray foraging optimization in feature selection problem," *Neural Computing and Applications*, vol. 33, no. 17, pp. 11027–11041, Sep. 2021, doi: 10.1007/s00521-020-05560-9.

BIOGRAPHIES OF AUTHORS

Jomana Yousef Khaseeb is a PhD. student in the Faculty of Computers and Information at Menoufia University, Egypt. She received her M.Sc. degree in Computer Science from Al-Quds University, Palestine, in 2010. She received the B.Sc. degree in Computer Science from Birzeit University, Palestine, in 2002. She works as a lecturer in Applied Computing Department, Palestine Technical University, Palestine. Her research interests include feature selection, metaheuristic algorithms, and classifications. She can be contacted at Jumanah.khaseeb@ptuk.edu.ps.

Arabi Keshk received the B.Sc. degree in Electronic Engineering and the M.Sc. degree in Computer Science and Engineering from the Faculty of Electronic Engineering, Menoufia University, in 1987 and 1995, respectively. He received his Ph.D. degree in Electronic Engineering from Osaka University, Japan, in 2001. His research interests include software engineering, distributed systems, cloud computing, machine learning, the IoT, big data analytics, and bioinformatics. He can be contacted at arabikeshk@yahoo.com.

Anas Youssef is an assistant professor in the Faculty of Computers and Information at Menoufia University, Egypt. He received his PhD degree in Electrical and Computer Engineering, specialization software performance engineering, from University of Calgary, Alberta, Canada. He received his M.Sc. degree in Computer Science from Menoufia University, Egypt. He received his B.Sc. degree in Computer Engineering from Cairo University, Egypt. His research interests include the following fields: resource management techniques in cloud computing; software performance modeling and evaluation; performance evaluation of parallel computer architectures; security in mobile computing systems; feature selection and classification of text data; and Arabic text mining. He published a set of research papers in prestigious international conferences and journals in the fields mentioned above. Anas Youssef has been teaching a wide variety of undergraduate and postgraduate university courses in different universities in Egypt and Canada. These universities include Menoufia University in Egypt, Cairo University in Egypt, Zewail City of Science and Technology in Egypt, American University in Cairo (AUC) in Egypt, Egyptian E-learning University (EELU) in Egypt and University of Calgary (UofC) in Calgary, Alberta, Canada. He can be contacted at anas.youssef@ci.menofia.edu.eg.