Educational Administration: Theory and Practice
2024, 30(1), 516-528
ISSN: 2148-2403
https://kuey.net/

Research Article

Hybrid Gradient Descent Grey Wolf Optimizer for Cloud Workload Balancing with Optimal Feature Selection using Reinforcement Learning

Prateek Aggarwal[1*], Gouri Sankar Mishra[2], Pradeep Kumar Mishra[3], Aditya Kumar[4]

[*1,3,4]Department of Computer Science and Engineering, SSET, Sharda University, Greater NOIDA, U.P., India.
prateekagarwal1963@gmail.com, pradeepkumar.mishra@sharda.ac.in, aditya.kumar1@sharda.ac.in
[2]School of Computer Science and Application, SSET, Sharda University, Greater NOIDA, U.P., India.
gourisankar.mishra@sharda.ac.in

Citation: Prateek Aggarwal, et al(2024) Hybrid Gradient Descent Grey Wolf Optimizer for Cloud Workload Balancing with Optimal Feature Selection using Reinforcement Learning, *Educational Administration: Theory and Practice*, *30*(1), 516-528
Doi: 10.53555/kuey.v30i1.4674

ARTICLE INFO	ABSTRACT

Method of reducing the elements from a dataset by removing irrelevant, redundant, and randomly selected features which is called feature selection. It aims to reduce training time and improve data quality especially for big and complex datasets. This study introduces an optimizer for feature selection problems by combining the metaheuristic algorithm called the grey wolf optimizer with gradient descent algorithm. The proposed approach outperformed the original grey wolf optimizer on various test functions and showed promising results on clinical datasets from the UCI machine-learning repository. It suggests potential by enhancing feature selection techniques in data analysis.

***Keywords*:** Grey Wolf Optimizer; Cloud Workload Balancing; Reinforcement Learning;

1. Introduction

Cloud Workload, particularly in converting aggregated data into useful information and strategies. It focuses on the need of tailored techniques and additional processing constraints. The resolution of cloud-related issues often requires a specific response within predetermined time and may involve optimization techniques to find optimal solutions[1]. Metaheuristic algorithms are highlighted for their effectiveness in guiding the search process towards optimal solutions for example - swarm intelligence algorithms that mimic insect behavior. Objective functions, both static and dynamic, which plays a crucial role in describing metaheuristics algorithm, especially in adapting to new restrictions and changes in the explorative space[2].
Different types of algorithms, which is nature-inspired ones like genetic algorithms, whale algorithms and others, which are discussed for their ability to follow their natural processes and solve optimization problems. The balance between power and broadening function capabilities in executing metaheuristic algorithms are also addressed, emphasizing the importance of feature selection and adaptable concepts for improved performance[3]. This study further explores various data mechanisms and classification techniques, which includes covering methods, to optimize performance and reduce false positive rates. It suggests the application of grey wolf algorithms as solutions for selection challenges in cloud dataset.
Overall, the complexity is processed by optimizing cloud data set, the role of metaheuristic algorithm and feature selection is used to enhanced performance in predictive techniques and decision-making processes.

2. Literature Review

Grey Wolf Optimization (GWO) techniques from 2015 to 2019, includes - Matched GWO, Hybrid GWO, Wild GWO and a GWO approach for Parkinson's Disease[4]. These methods involve equal data factors which incorporates for speed, and a feature assurance process driven by man-made intelligence models. This study evaluates the accuracy and efficiency of KNN, random forests, and decision tree models across four datasets concluding that random forests outperform KNN, particularly in decision-making strategies resembling a cuttlefish estimation[6] [7] [8].

2.1. Inferences Drawn

Grey Wolf optimizer and its variations in optimization processes and parameter selections. It emphasizes on the optimizer compatibility, simplicity and few constraints with other optimization agents. Additionally, it suggests that the covering method is a robust feature selection technique, with the KNN classifier offering low error rates without overfitting issues.

2.2.Grey Wolf Optimizer

Mirjalili et al. (2014) introduced GWO Enhancer, a metaheuristic algorithm which inspired by the hunting behaviors of gray wolves. This algorithm simulates the roles within a wolf pack, including the Alpha (α), Beta (β), Delta (δ), and Omega (ω) wolves, each with distinct responsibility. The Alphas lead the pack in decision-making and hunting strategies, Betas support them and may inherit the Alpha role, Delta includes various support roles, and Omegas are subordinate[9].

This social hierarchy is reflected in the algorithm approach for problem-solving, where leaders coordinate actions, scouts identify opportunities, and the entire pack converges to attack to get the solution. The algorithm's process is modeled in three steps, mirroring the stages of a wolf pack's hunt.

$$\vec{M} = \left|\vec{L}.\overrightarrow{X_u}(s) - \vec{X}(s)\right|, \qquad (1)$$

$$\vec{X}(s+1) = \overrightarrow{X_u}(s) - \vec{J}.\vec{M}. \qquad (2)$$

The environmental elements of the prey are shown as in Condition (1)

M ⃗ not entirely settled as the distance between the continuous wolf vector X ⃗(r) and the prey (X_u) ⃗.

X ⃗(s+1) is the accompanying worth of X, and J ⃗and L ⃗are unpredictable vectors of viewpoints identical to the components of X made from (q_1) ⃗and (q_2) ⃗of the range [0,1] and with a scope of [0,2]

$$\vec{J} = 2\vec{j} \cdot \overrightarrow{q_1} - \vec{j}, \qquad (3)$$

$$\vec{L} = 2 \cdot \overrightarrow{q_2}. \qquad (4)$$

j ⃗decreases from 2 to 0 that models the orbiting of the prey, as the emphasis counts up, as communicated in condition (5) beneath:

$$\vec{j} = 2 - \left(\frac{2 \times \text{iter}}{\text{Max}_{\text{iter}}}\right). \qquad (5)$$

2.3. Hunting the Prey

Hunting maps examine the request space as driven by the Alpha, Beta, and Delta. Showing this incorporates getting the Alpha, Beta, and Delta search experts from the pack by differentiating the health regards and picking the principal trained professionals[10]. The Omega positions are then invigorated by the primary wolves.

$$\overrightarrow{M_\alpha} = \left|\overrightarrow{L_1} \cdot \overrightarrow{X_\alpha} - \vec{X}\right|, \qquad (6a)$$

$$\overrightarrow{M_\beta} = \left|\overrightarrow{L_2} \cdot \overrightarrow{X_\beta} - \vec{X}\right|, \qquad (6b)$$

$$\overrightarrow{M_\delta} = \left|\overrightarrow{L_3} \cdot \overrightarrow{X_\delta} - \vec{X}\right|, \qquad (6c)$$

$$\overrightarrow{X_1} = \overrightarrow{X_\alpha} \cdot \overrightarrow{J_1} \cdot (\overrightarrow{M_\alpha}), \qquad (7a)$$

$$\overrightarrow{X_2} = \overrightarrow{X_\beta} \cdot \overrightarrow{J_2} \cdot (\overrightarrow{M_\beta}), \qquad (7b)$$

$$\overrightarrow{X_3} = \overrightarrow{X_\delta} \cdot \overrightarrow{J_3} \cdot (\overrightarrow{M_\delta}), \qquad (7c)$$

$$\vec{X}(s+1) = \frac{\overrightarrow{X_1} + \overrightarrow{X_2} + \overrightarrow{X_3}}{3}, \qquad (8)$$

2.4. Attacking And Searching the Prey

Whenever the prey stops moving, the pack goes for the last final knockout, sending all of the experts towards the prey from different places.

The numerical model of these is subject to vector $\vec{A}$. At the point when the outright worth of vector $\vec{A}$ is under 1, the pursuit experts meet towards the prey when they search the space[11].

2.5. Feature Selection

Feature decision is the most well-known approach to lessening the number of components in a dataset by dispensing with overabundance, pointless, and randomly class-redressed data features. In doing so, a model is prepared to grow its precision as well as diminish overfitting and plan time by utilizing the ideal subset. It has applications in many fields, including text mining, picture dealing, clinical investigation, and issue finding[12]. The overall system of element choice includes four critical stages , as displayed in Fig 2:

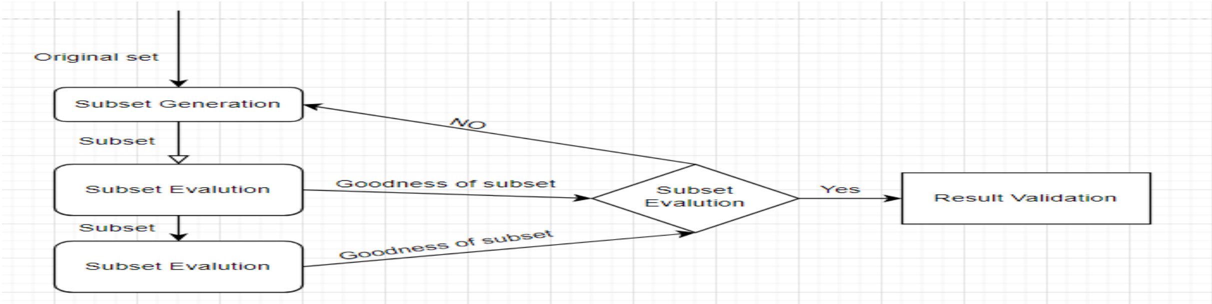

Fig.2 Feature Selection Process

The feature selection process involves 4 key stages:
1.Subset Generation: It Begins with a heuristic search like bidirectional, forward or backward to build an initial feature subset.
2.Subset Evaluation: Determines the subset "fitness" based on a defined rule which possibly tied to the mining algorithm.
3. Stopping Rule: Decides when to end the feature selection process based on criteria like model performance, minimum/maximum feature count, or convergence.
4. Result Validation: It selects and evaluate the best feature subset, using either filter methods based on wrapper methods or performance metrics which evaluated by an unknown modeling algorithm .
Wrapper methods, though slower, ensure balanced performance in the chosen model, making them preferred for optimal results. Hybrid techniques combine aspects of both filter and wrapper methods to fine-tune models during training.

2.6. Gradient Descent Iterative Stochastic

Conceptualizing an objective capacity as a dark scene, the foundation of this domain may be pushed toward little by little, industriously moving against the tendency of the region[13].
Tendency dive does this as an iterative update pattern of the position θ.

$$\theta = \theta - \eta\nabla_\theta A\left(\theta; x^{(i)}; y^{(i)}\right). \qquad (9)$$

For a function whose objective function is defined as $\left(\theta; x^{(i)}; y^{(i)}\right)$, whose partial derivative with respect to each parameter of $x^{(i)}$is $\nabla_\theta A\left(\theta; x^{(i)}; y^{(i)}\right)$, the core equation of gradient descent is shown in equation (9).
The computation circles a set number of times (most outrageous cycles); at each accentuation, the value of θ is invigorated by working out the fragmentary subordinate of the objective capacity concerning the limits of the data and deducting this value from θ . In doing so, the computation assesses the effect each limit has on the objective capacity and uses this information to control the heading and speed of get over in the pursuit space. The variable η controls the learning rate, avoiding the two movements about a base achieved by tremendous potential gains of the mostly subordinate and slow mix rates achieved by low fragmentary auxiliary characteristics.

Pseudocode 2.1- Gradient Descent Iterative Stochastic.

1. Start
2. Heedlessly instate the position vector and pick a sensible worth of set size
3. While the best number of emphases isn't outperformed
a. Assess the fractional subsidiary of the goal capability as for the components of the pos. vector
b. Update the pos. vector as indicated by condition (9)
4. Stop
5. Arrangement == θ
6. Stop

For a clear capacity space, the partial subordinate encodes the heading wherein the closest close by least exists. As the capacity extensions in unpredictability with various close by minima exist, the value of the mostly subordinate as a vector in the capacity space could point at the weighted typical of the neighborhood minima. To avoid this outcome, the computation is ordinarily run on different events, with each basic starting region randomized. This computation is striking in execution in mechanized thinking, expressly in backslide and cerebrum associations. With a mathematically deduced deficient auxiliary, the computation can achieve a high mix rate, and with various sporadic presentations, it sidesteps neighborhood minima. One of its gigantic deficiencies is that when the mostly subordinate assessment is heightened, the computationally multifaceted nature of the general cycle is extended.

2.7. Reinforcement Learning

Making decisions to maximize rewards in specific condition. It is used in various programs to determine the best course of action. It uses labeled training data, and it relies on a reward system. Reinforcement learning algorithms learn by receiving input after each action, determining the effectiveness of their choices. It focuses on maximizing rewards, collecting data through trial and error, and adjusting strategies based on feedback. This self-teaching approach is valuable for autonomous systems by making instant decisions without human involvement, which aims to optimize outcomes through continual adaptation[14].

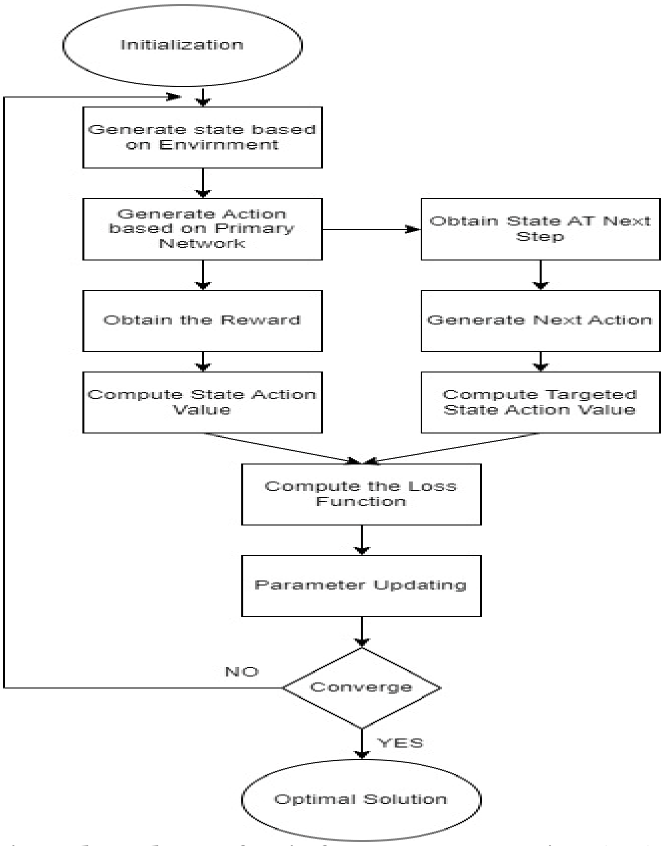

Fig 3 Flow chart of Reinforcement Learning (RL)

3. Proposed Methodology

3.1 Grey Wolf Optimizer Hybrid Gradient Descent (GWOHGD)

The slope plunge of a wolf position may be conceptualized as the heading from which the wolf seems to smell the prey as of now. The primary wolves would then smell the air, and each picks a representative among the most over-the-top unpleasantly horrendous performers from the pack and calls them to itself, then, at that point, trains it to "follow" the scent. The crucial execution is given in Pseudocode 2, under.

Pseudocode 3.1 - GWOHGD.

1. Begin
2. Arbitrarily introduce all wolves inside as far as possible
3. Assess the wellness upsides of all wolves and sort out rising requests.
a. Set the alpha wolf as the most imperative aspect of wellbeing.
b. Set the beta wolf as the second most significant wellbeing regard
c. Set the delta wolf as the third most raised health regard.
4. While the best number of accentuations isn't outperforming,
a. For each wolf,
i. Assess J and L utilizing conditions (3) and (4).
ii. Assess each of the 3 upsides of M utilizing conditions ((6a), (6b) and (6c)).
iii. Assess X1, X2 and X3 utilizing conditions ((7a), (7b) and (7c))
iv. Assess the new positions utilizing condition (8).
b. Stop
c. Evaluate the partial auxiliary of the alpha, beta, and delta wolves from condition (10)
d. Update the last 3 health regard wolves using condition (9) with the update regions as the alpha, beta, and delta wolves.
c. Evaluate the wellbeing potential gain of the wolves.
5. Stop
6. Arrangement == alpha wolf

The partial is not entirely set in stone as a conjecture since the objective capacity is taken as dark. This is done by picking a phase variable that merits remembering for the wolf position in every perspective as well as being deducted. The health potential gains of these two new are not set in stone and deducted from each other, all in all, apportioned by the degree of regard. This is the center restricted differentiation for auxiliary appraisals.

(δJ((x)) ̃J/(δx_i)= (A(x_1,x_2,…,(x_i+ r),…,x_n)- A(x_1,x_2,…,(x_i- r),…,x_n))/r , (10)

3.2 GWOHGD (Binary Version)

The equal version changes the proposed persevering computation for feature decision while taking advantage of the restricted chase space of a dataset: each region in space either recollecting or banishing a part from thought for the model[15].

As a multi objective component assurance issue, the health capacity changes with this as a weighted measure of the two confining game plan requirements, less picked features, and low error rates. An enormous change is in the assessment of the wellbeing capacity. Since there are two objectives of the health capacity: to help the computation look through out game plans with less components, a "cost" is given out to the amount of features and added to the wellbeing ability. Still up in the air as beneath:

fit= ∅Err+ θ R/M , (11)

where ∅ balances the bumble of the fitted model, Fizzle, and θ=(1-∅) changes the extent of the amount of picked features R to the total number of components M.

The progressions to the mostly auxiliary for a matched pursuit space are shown in Pseudocode 3 under. For the fragmentary subordinate, every component document goes through the equal NOT action, and this value is deducted from the main mostly differential.

Pseudocode 3.2 - Halfway subordinate double form.

1- Ability fragmentary auxiliary
2- Pass in: wolf_position, fitness_function
3- Set partial_derivatives as the vector as zeros
4- For every part of the wolf_position
a- Set new_position as Cancel not on the opportunity that Wolf_position incorporates.
b- Set the new_fitness regard as Call fitness_function for new_position
c- Set partial_derivative record I as the qualification of new_fitness and health of wolf_position
5- End for
6-Drop: midway subordinates
7- End

At the point when the wolf positions are mostly still up in the air, this information is utilized in reviving the wolf positions through the change by fragmentary auxiliary capacities. Three executions were used as follows:
(1) With this execution, shown in Pseudocode 4, the mostly subordinate is used as a fundamental cutoff marker with higher auxiliaries provoking lower feature record inversion rates. The higher the mostly subordinate, the higher the chance of changing the wolves' worth at the partial auxiliary rundown
(2) In this execution, shown in Pseudocode 5, variable α is used to merge examination and cheating stages during the iterative chase process. It does this by changing the breaking point for the Wolf Record Change.
(3) With this, shown in Pseudocode 6, the sigmoid capacity is used to design the partial subordinate as far as possible space with a normalized mostly auxiliary while at this point solidifying variable α for examination and misleading. This ensures the implantation of inversion regardless, when there is apparently no significant update heading for the wolf.

Pseudocode 3.3- Execution 1.

1- Ability changes with inadequate auxiliary
2 - Pass in: Wolf_pos, auxiliaries, probability
3 - Set the robability vector aas a caled subordinate vector copied by probability
4 - Set selected features as probability vectors to take a gander at against a created sporadic number vector
5 - Set new_pos as Call XOR of wolf_pos and selected_features
6 - Drop: new_pos

Pseudocode 3.4 - Execution 2.

1- Capacity c changeswith deficient subordinate
2- Pass in: wolf_pos, subordinates, probability edge, a
3- Set load as 0.4*a + 0.1
4- Normalize the mostly subordinate
5- Process the sigmoid of the normalized fragmentary subordinate and set to variable sig
6- Use the probability cutoff and weight to pick the component records to change
7- Figure out the new wolf position and pass back.

Pseudocode 3.5 - Execution 3.

1- Ability changes with mostly subordinate
2- Pass in: wolf_pos, auxiliaries, probability edge, a
3- Set load as 0.4*a + 0.1
4- Set the expoilation_probabilities as arranged auxiliary on the sigmoid twist centered at nothing
5- Set the exploration_probalilities as the extent of features decided to amount to features for the picked components and 1-this for negative
6- Set selected_features as how much weight expoilation_probabilities and 1 - weight exploration_probalilities copied by probability_threshold took a gander at against a vector of indiscriminately made numbers
With change getting done, the most clearly horrendous working out of wolves' positions is invigorated as the modified positions are surveyed. The three most appallingly horrendous performing wolves were chosen to revive the circumstances of the alpha, beta, and delta wolves.

4. Experimentation and Result Analysis

The estimations were done in MATLAB. The steady variation was first pursued for reasonableness and gave information on its show in different determined ability progression issues. The matched variation went through various changes to additionally foster execution[16].

4.1. GWOHGD

As a ceaseless worth streamlining capability, the cross breed calculation was tried on 12 benchmark capabilities. Each capability was run multiple times, and the assembly bend and last worth were recorded[17].

4.1.1.Test Capabilities

12 benchmark capabilities were utilized to look at the presentation of the GWOHGD against GWO[18]. They are isolated into three sorts of capabilities: Unimodal Capabilities (M1-M4). Double-dealing examination for really looking at the abuse capacity of the enhancer (Figures 4-7).

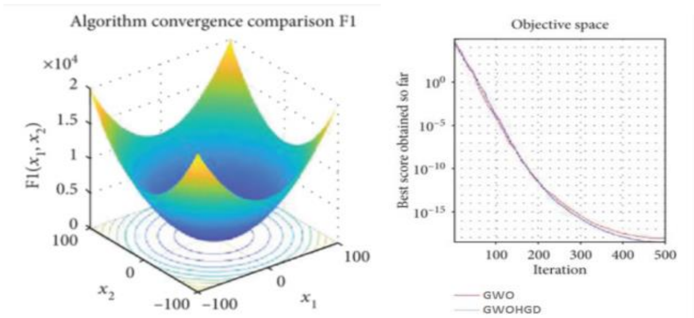

Fig 4 Blend outline of unimodal benchmark ability (M1).

GWO shows Grey wolf headway; GWOHGD exhibits faint wolf smoothing out specialist creamer point plunge.

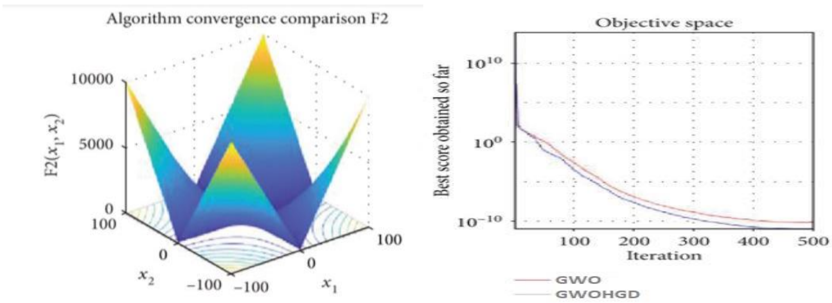

Fig 5 Intermingling chart of unimodal benchmark capability (M2).

GWO demonstrates grey wolf enhancement; GWOHGD shows grey wolf analyzer half and half angle plunge.

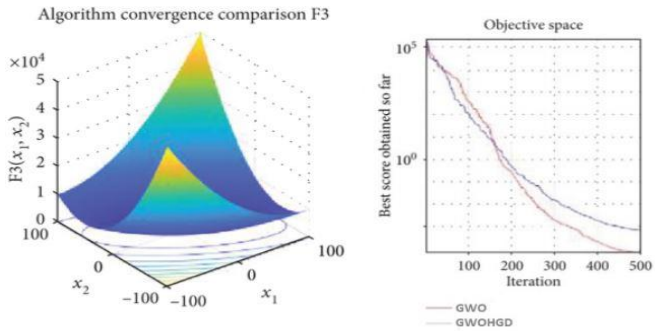

Fig 6 Intermingling chart of unimodal benchmark capability (M3).

GWO demonstrates grey wolf enhancement; GWOHGD shows grey wolf analyzer half and half angle plunge.

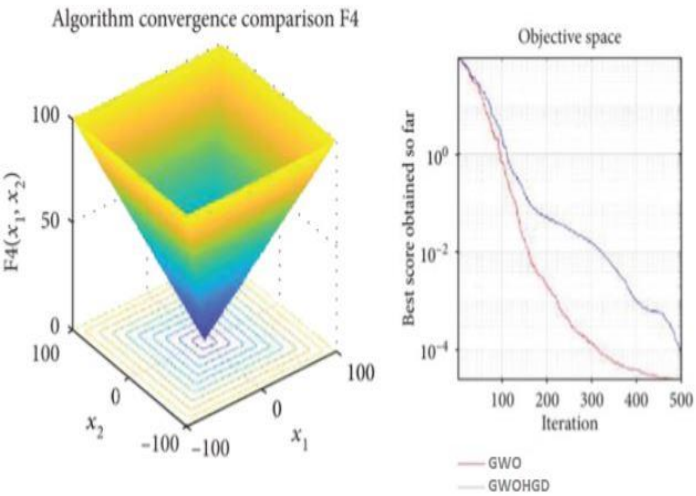

Fig 7Assembly diagram of unimodal benchmark capability (M4).

GWO shows grey wolf advancement; GWOHGD demonstrates grey wolf analyzer crossover angle plunge. Multimodal Capabilities (M5-M8). Investigation examination for really looking at the investigation capacity of the enhancer (Fig. 8-11)

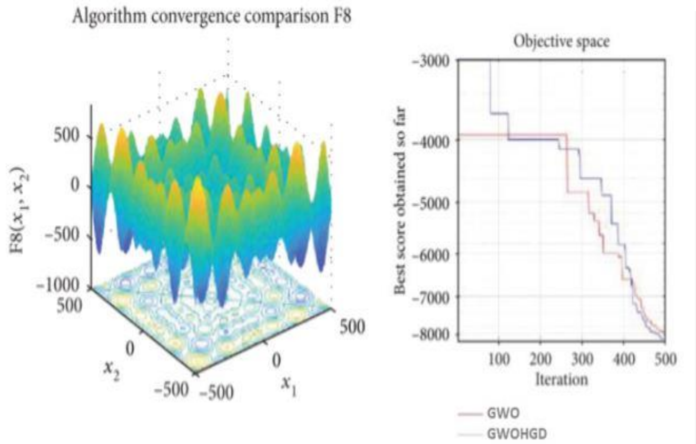

Fig 8 Assembly diagram of multimodal benchmark capability (M5).

GWO shows grey wolf improvement; GWOHGD demonstrates grey wolf analyzer half breed inclination plunge.

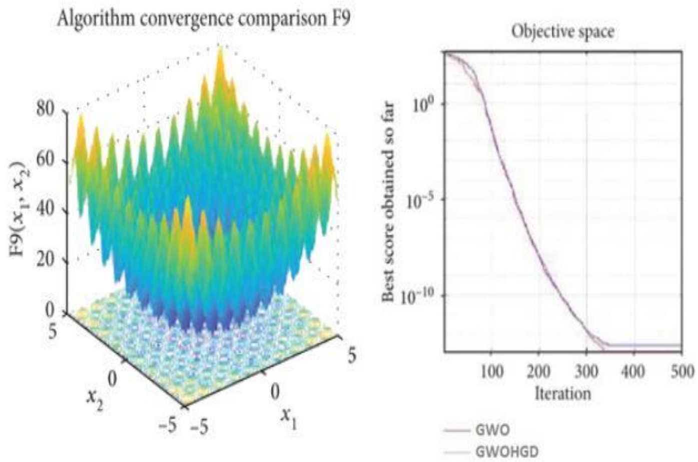

Fig 9 Combination chart of multimodal benchmark capability (M6).

GWO demonstrates grey wolf advancement; GWOHGD shows grey wolf analyzer mixture angle plunge.

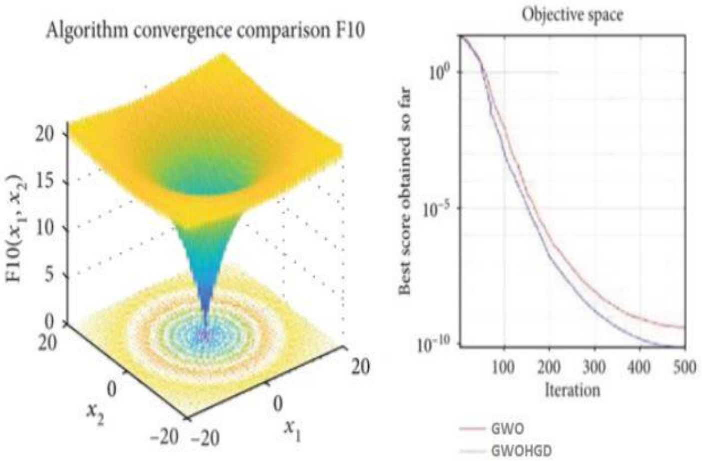

Fig 10 Intermingling diagram of multimodal benchmark capability (M7).

GWO shows grey wolf streamlining; GWOHGD demonstrates grey wolf analyzer half and half inclination plunge.

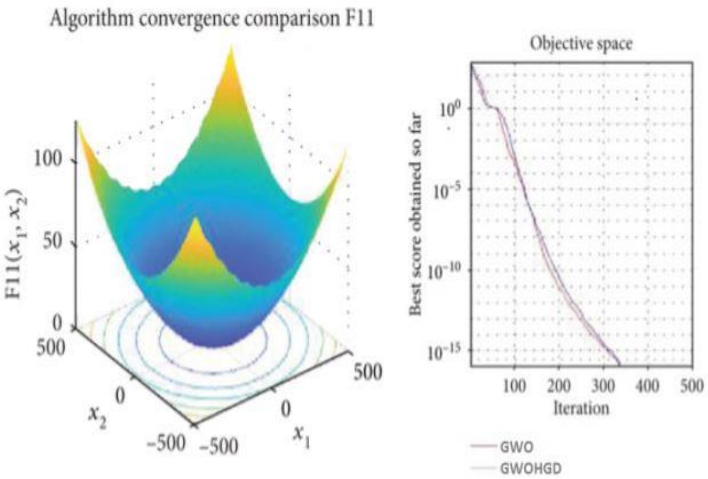

Fig 11 Assembly chart of multimodal benchmark capability (M8).

GWO shows grey wolf advancement; GWOHGD demonstrates mixture angle plummet grey wolf enhancer. Fixed-Aspect Multimodal Capabilities (M9-M12). For examination of the investigation capacity of the calculation on account of fixed-aspect streamlining issues (Fig. 12–15).

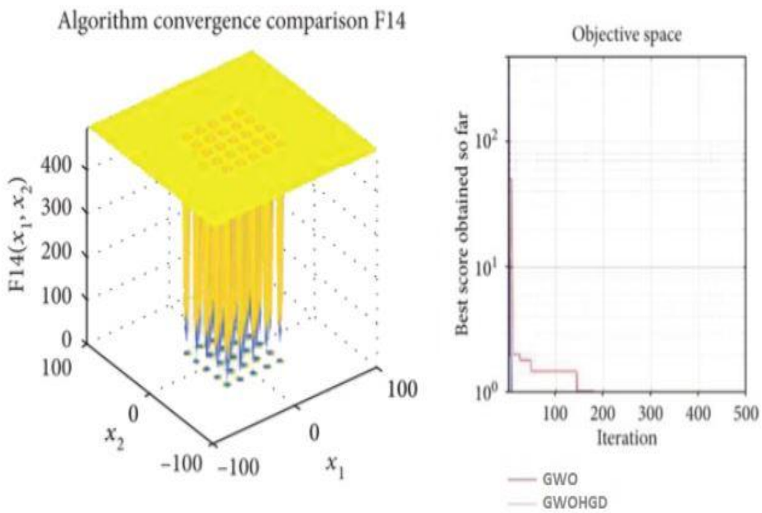

Fig 12 Intermingling chart of fixed-aspect multimodal benchmark capability (M9).

GWO shows grey wolf improvement; GWOHGD demonstrates half breed angle plummet grey wolf streamlining agent.

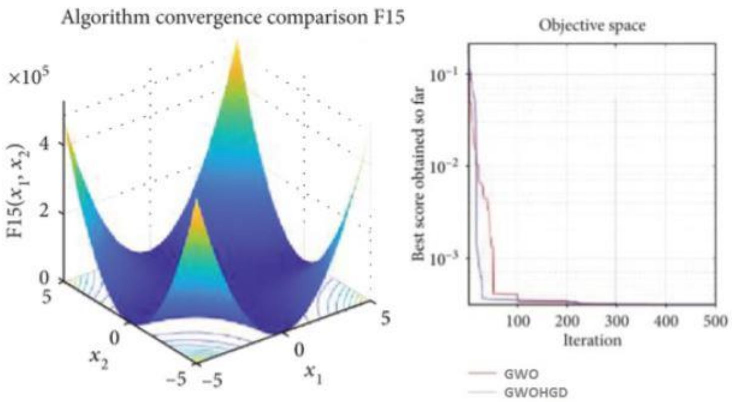

Fig 13 Assembly chart of fixed-aspect multimodal benchmark capability (F10).

GWO demonstrates grey wolf streamlining; GWOHGD shows mixture inclination drop grey wolf analyzer.

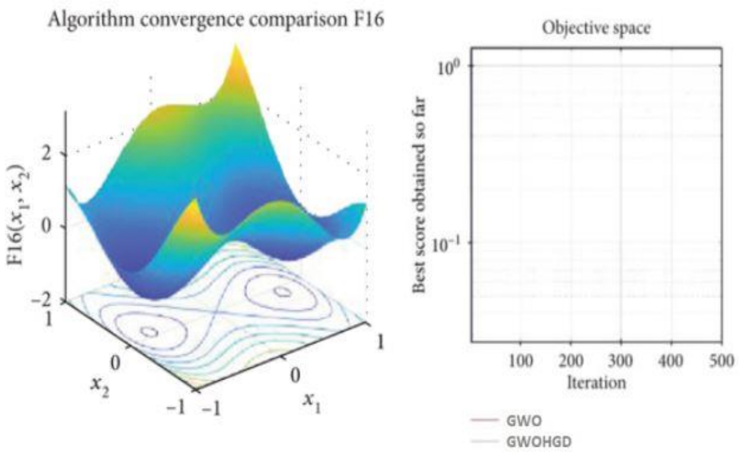

Fig 14 Assembly diagram of fixed-aspect multimodal benchmark capability (M11).

GWO demonstrates grey wolf enhancement; GWOHGD shows half and half angle plummet grey wolf streamlining agent.

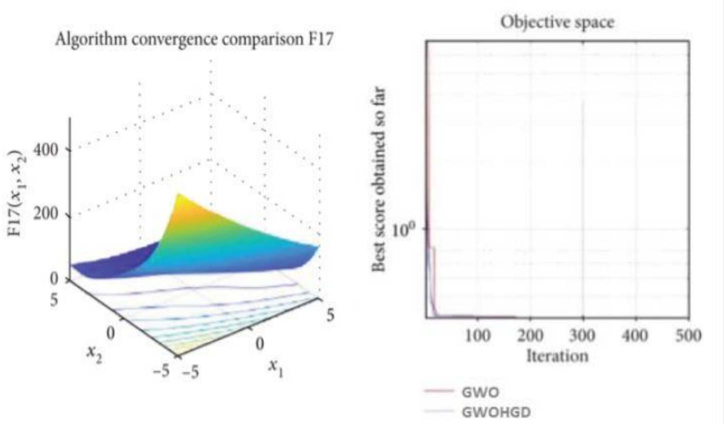

Fig 15 Union diagram of fixed-aspect multimodal benchmark capability (M12).

GWO shows grey wolf improvement; GWOHGD demonstrates half breed angle plummet grey wolf streamlining agent grey wolf enhancer.

4.1.2.Parameter Settings

(i) Number of analyses =600
(ii) Maximum cycle = 600
(iii) Number of wolves =12
(iv) Learning rate η=0.003 .
This is a variable for the partial auxiliary, portraying the position update rate. A high learning rate extends the blend rate while growing movements around a close by least.
(v) Step size s=1 E-15. For inadequate auxiliary assessment, a high worth declines the precision of still up in the air
The limits were tuned iteratively to achieve needed results.

4.1.3. Evaluation Metrics

(1)Standard deviation

$$\sigma^j = \sqrt{\frac{\sum_{i=0}^{n}(R_j^i - \mu)^2}{B}} \qquad (12)$$

This is the extent of the closeness between different game plans. A raised prerequisite deviation shows gigantic changes in the course of action as the capacity runs through various events. A low distinction exhibits a to some degree static plan free of the amount of re-initializations.
(2) Average of solutions

$$\mu^j = \frac{\sum_{i=0}^{n} R_j^i}{B} \qquad (13)$$

(3)Minimum solution

$$\text{Minimum} = \min(R^j). \qquad (14)$$

This is the most reduced worth of the wellness esteem accomplished over the complete number of reiterations.
(4)-Timing
The MATLAB timing capability "timeit" is utilized. The calculations are planned on what amount of time they require for on each capability, from calling to returning the arrangement.

4.2. GWOHGD (Binary Version)

4.2.1 Datasets

A. A. From the UCI simulated intelligence file, 6 clinical datasets were picked and used to test GWOHGD in feature assurance applications against BGWO executions 1 and 2 as well as BGWOPSO. Table 1 shows the specific datasets used and looks at feature numbers and tests.

Table 1 Datasets for evaluating binary GWOHGD.

	No. of instances	No. of features
Bosom Disease Wisconsin (Analytic)	682.8	36
Bosom Disease Wisconsin (Unique)	838.8	10.8
SPECT Heart	320.4	26.4
Stat log	324	15.6
Heart Disease (Coronary Artery Disease)	363.6	16.8
Lymphographic	177.6	21.6

i) Bosom Disease Wisconsin (Demonstrative) - Features are enrolled from a digitized image of a fine needle pull (FNA) of a chest mass. They portray the properties of the cell centers present in the image.
ii) Chest Illness Wisconsin (Exceptional) - Some of its components are pack thickness, consistency of cell shape, consistency of cell size, and single epithelial cell size, among others.
iii) SPECT HEART - The dataset depicts diagnosing cardiovascular Single Photon Radiation Handled Tomography (SPECT) pictures. All of the patients are requested into two classes: normal and uncommon.
iv) Statlog (heart) - The class is accumulated as either the nonappearance or the presence of a coronary ailment. The data integrates patient information and secondary effects as well as clinical preliminary outcomes. It consolidates features, for instance, age, sex, chest torture type, resting circulatory strain, cholesterol, and fasting glucose.
v) Lymphography - The dataset is collected into 4 classes: common find, metastases, affront lymph, and fibrosis. The components are characteristics of the center points of the patient, including shape, leaves, and extravagates. A portion of the classes are disproportionally tended to, influencing the distribution of the dataset.
vi) Cleveland Coronary Disease (Coronary Stockpile Course Disorder) - The dataset consolidates patient information, for instance, age, sex, fasting glucose, and cholesterol, intending to ensure patient assurance.

4.2.2. Parameter Settings
(i) Number of wolves: 12
(ii) Maximum number of emphases = 60
(iii) Number of tests: 12
(iv) Random Wolf Presentation Edge = 0.3. This variable sets the amount of beginning picked features with better calibers and lower initial picked features
(v) Partially subordinate change edge = 0.9 . This fills in as an adjustment of edge position to change the chance of inversion of a part
(vi) Limits of misuse versus examination weight = [0.1,0.9] . This defines the boundaries for the probability of examination and cheating. The limits were tuned iteratively to achieve the needed results.

4.2.3 Feature Selection
The component assurance procedure used was the covering based method with the following presets:
(i) Objective ability is mostly auxiliary, as conveyed in condition (10)
(ii) The search procedure is GWOHGD (inconsistent bidirectional)
(iii) Modeling estimation is KNN with a K worth = square base of the investigations
(iv) The distance calculation capacity is the Euclidean distance
(v) F-overlay is 6
(vi) An external classifier is the assist vector in machining
(a) Standardize is legitimate
(b) Kernel capacity is RBF
(c) The kernel scale is auto
(vii) Data separating ability cv-section and number of parts =12 for 2 class datasets and 2 for classes more than 2 with the transport skewed The covering system was picked for its unparalleled display, as imparted in the composing study. The KNN classifier was moreover used for health evaluation because of its authentic tendency in feature assurance, unequivocally with GWO varieties.

4.2.4. Assessment Measurements
The measures utilized once an answer was accomplished were as per the following:

i)Average grouping precision This is a proportion of the legitimacy of a model's expectations.

$$\text{Avearage accuracy} = \frac{1}{C}\sum_{i=0}^{n} \text{Acc}^{i} \qquad (15)$$

ii)Normal number of chosen highlights
iii) Normal well-being values
iv)Sensitivity
This is the extent of precisely expected positive cases to amount to positive cases.

$$\text{Sensitivity} = \frac{SU}{SU + FB} \qquad (16)$$

v)Precision
This is the proportion of restoratively anticipated positive cases to add up to anticipated positive cases.

$$\text{Precision} = \frac{SU}{SF + FU} \qquad (17)$$

vi)F-measure
This is the symphonious normal of accuracy and responsiveness.

$$\mathrm{F-measure} = 2\frac{\mathrm{Precision} * \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}, \quad (18)$$

vii) One-way ANOVA test on the quantity of elements chose by every calculation. The worth was gotten from the MATLAB capability "anova1"

4.3. Correlation of GWOHGD versus GWO

HGDGWO is proposed to differentiated against GWO and manages to achieve higher typical health values for 5 of the 8 wellbeing works yet achieves a lower least health regard in 9 of the 12, as seen from Table 2.

Table 2 Correlation of GWOHGD versus GWO on the standard deviation, normal arrangement worth, and least arrangement esteem.

	GWO			HGDGWO		
	Std	Avg	Min	Std	Avg	Min
M1	7.23353-15	15-Mar	2.10-18	5.104704-16	2.24676-16	6.85772-19
M2	10.20704-10	12.28-10	13.69-11	16.45530-11	2.14402-10	19.02394-12
M3	3.9411607	0.8819	14.10-05	18.534908	5.048532	0.00146
M4	0.0022	0.00194	4.5-05	1.254	2.2884	18.22584-05
M5	1729.364	-11138	-15939.6	1509.8108	-10662.9	-16401.4
M6	17.009538	14.94272	2.28-13	9.8975325	10.58192	4.54748-13
M7	8.578-09	12.2-09	8.58-10	2.3605-09	2.81919-09	15.24416-11
M8	0.0319496	0.0197	2.22-16	0.0372313	0.030115	0
M9	9.0207949	13.02434	1.996	9.682958	15.07511	1.996008
M10	0.0180751	0.01056	0.00062	0.0194616	0.012374	0.000615
M11	12.45262-07	-2.06326	-2.06326	2.26176-07	-2.06326	-2.06326
M12	0.0006305	0.79586	0.79578	3.3668-05	0.795796	0.795775

The enhancer likewise expands the typical calculation time by a component of 5.6, as seen from Table 3.

Table 3 Correlation of GWOHGD versus GWO on runtime in a flash.

	Timing in seconds	
	GWO	GWOHGD
M1	0.169176	0.677548296
M2	0.184944	0.876622776
M3	0.485352	7.072135176
M4	0.173112	0.943126776
M5	0.157152	0.863391096
M6	0.144888	0.643143816
M7	0.151368	0.775612776
M8	0.16572	0.985311336
M9	0.886824	1.295330136
M10	0.120072	0.223459416
M11	0.1368	0.280348536
M12	0.175416	0.277195416

Method proposed in the study shows a high association rate for specific functions as shown in figures. It performs great in certain methods compared to the main method particularly in method 3, 4 and 12 as shown in corresponding figures. However, it is closely matches the main method performance in several other method. The HGDGWO method demonstrates effectiveness in exploiting optimal solutions, especially in multimodal functions.
Hybrid function also increases the average time significantly, as anticipated due to increased algorithmic complexity.

6. Future Work and Conclusion

Existing wolf optimization methods is used to find the prey directions with the help of designed Blend Point Fall Faint Wolf Analyzer. It achieves better results in feature selection problems, surpassing the BGWO2 algorithm in multiple datasets. The proposed algorithm's accuracy is high lighted, especially in data subset, though further enhancements are suggested.
These include adding a memory module to reduce computational load and implementing a method to prevent stagnation in local optima. Grouping features into packs and adjusting their partial derivatives are also proposed to streamline calculations.

References

1. C. Voudouris, E. Tsang, and A. Alsheddy, Guided local search, Springer, Boston, MA, 2010.
2. H. R. Lourenco, O. C. Martin, and T. Stutzle, "Iterated local search," 2001, February 2021.
3. Kumar J, Rani A, Dhurandher SK (2020) Convergence of user and service provider perspectives in mobile cloud computing environment: taxonomy and challenges. Int J ommunSyst 33(18):e4636
4. Sefati S, Abdi M, Ghafari A (2021) Cluster-based data transmission scheme in wireless sensor networks using black hole and ant colony algorithms. Int J Commun Syst.
5. Hayyolalam V, Pourghebleh B, Kazem AAP, Ghafari A (2019) Exploring the state-of-the-art service composition approachesin cloud manufacturing systemsto enhance upcoming techniques. Int J AdvManufTechnol 105(1):471–498
6. Alicherry M, Lakshman T (2013) Optimizing data access latencies in cloud systems by intelligent virtual machine placement. In: 2013 Proceedings IEEE INFOCOM, 2013. IEEE, pp. 647–655
7. Nayak SC, Parida S, Tripathy C, Pati B, Panigrahi CR. Multicriteria decision-making techniques for avoiding similar task scheduling conflict in cloud computing. Int J Comm Syst. 2019;n/a:e4126.
8. Karam, Y., Baker, T., &Taleb-Bendiab, A. (2012), Intention-oriented modelling support for socio-technical driven elastic cloud applications. Paper presented at the 2012, International Conference on Innovations in Information Technology (IIT)
9. Albu A, Precup R-E, Teban T-A. Results and challenges of artificial neural networks used for decision-making and control in medical applications. FU Mech Eng. 2019;17(3):285-308
10. Chatterjee U. A study on efficient load balancing algorithms in cloud computing environment. International Journal of Current Engineering and Technology. 2013;3:1767-1770.
11. Tang, L., Pan, J.-S., Hu, Y., Ren, P., Tian, Y., & Zhao, H. (2015). A novel load balance algorithm for cloud computing. Paper presented at the International Conference on Genetic and Evolutionary Computing.
12. Yang M, Li Y, Jin D, Zeng L, Wu X, Vasilakos AV. Softwaredefined and virtualized future mobile and wireless networks: a survey. Mobile Network Appl. 2015;20(1):4-18.
13. Karahoca A, Karahoca D, Aksöz M. Examining intention to adopt to internet of things in healthcare technology products. Kybernetes. 2018;47(4):742-770.
14. Fahim, Y., Ben Lahmar, E., El Labrlji, E.H., Eddaoui, A.: The load balancing based on the estimated finish time of tasks in cloud computing. In: 2nd World Conference on Complex Systems (WCCS), pp. 594–598 (2014)
15. Madivi, R., Kamath, S.S.: An hybrid bio-inspired task scheduling algorithm in cloud environment. In: International Conference on Computing, Communication and Networking Technologies (ICCCNT), pp. 1 –7 (2014)
16. Domanal, S.G.R., Ram Mohana, G.: Load balancing in cloud computing using modified throttled algorithm. In: IEEE International Conference on Cloud Computing in Emerging Markets (CCEM), pp. 1–5 (2013)
17. Sharma, A., Peddoju, S.K.: Response time based load balancing in cloud computing. In: International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), pp. 1287–1293 (2014)
18. S. Mirjalili, "The ant lion optimizer," Advances in Engineering Software, vol. 83, pp. 80–98, 2015.