Fusion of Water Evaporation Optimization and Great Deluge: A Dynamic Approach for Benchmark Function Solving

Saman M. Almufti*1
1 Nawroz University, College of Science, Computer Science Department, Duhok, Iraq
Emails: Saman.Almofty@gmail.com

Abstract

The "Water Evaporation Optimization - Great Deluge" explores the synergy between the Water Evaporation Optimization Algorithm (WEOA) and the Great Deluge Algorithm (GDA) to create a novel fusion model. This research investigates the efficacy of combining these two powerful optimization techniques in addressing benchmark problems. The fusion model incorporates WEOA's dynamic exploration-exploitation dynamics and GDA's global search capabilities. By merging their strengths, the fusion model seeks to enhance convergence efficiency and solution quality. The study presents an experimental analysis of the fusion model's performance across a range of benchmark functions, evaluating its ability to escape local optima and converge towards global optima. The results provide insights into the effectiveness of the fusion model and its potential for addressing complex optimization challenges., a comprehensive performance analysis of the application of the proposed fusion model to a curated set of widely acknowledged benchmark functions, renowned for their role in evaluating the capabilities of optimization algorithms, is undertaken. By rigorously evaluating the convergence characteristics, solution quality, and computational efficiency of the algorithm, a thorough understanding of the strengths and limitations of WEOA is aimed to be provided. Through meticulous comparisons with established optimization techniques, illumination of the aptitude of WEOA in addressing diverse optimization challenges across a spectrum of problem landscapes is intended. The analytical insights, not only advancing the understanding of WEOA's applicability, but also furnishing valuable guidance for both researchers and practitioners in search of robust optimization methodologies, are proffered.

Keywords: Water Evaporation Optimization Algorithm; Metaheuristic Algorithms; Benchmark Functions; Fusion Model.

1. Introduction

The field of optimization has witnessed substantial advancements in recent years, driven by the development and application of metaheuristic algorithms. These algorithms offer versatile and powerful solutions to a wide range of complex optimization problems encountered across diverse domains, spanning from engineering and finance to artificial intelligence and logistics. Among these innovative algorithms is the Water Evaporation Optimization Algorithm (WEOA), a nature-inspired approach that draws its inspiration from the intricate process of water evaporation. The WEOA introduces a novel perspective on optimization by simulating the behavior of water molecules in a thermodynamic system to navigate complex solution spaces.

The promise of metaheuristic algorithms as powerful tools for addressing the complexities of optimization issues, has received attention. These algorithms, which draw inspiration from natural processes, social behaviors, and physical occurrences, offer a versatile method to solving complex optimization problems [1]. Prior research in the context of the PVDP has investigated various well-known metaheuristic approaches, including Particle Swarm Optimization(PSO), Genetic Algorithms (GAs) , Big Bang-Big Crunch (BB-BC), Cuckoo Search Algorithm (CSA), Artificial Bee Colony (ABC), and Vibrating Particles System (VPS)[2].

Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

In pursuit of enhancing optimization capabilities, researchers have explored hybrid approaches that combine the strengths of different algorithms. One such endeavor in this study involves the fusion of the Water Evaporation Optimization Algorithm (WEOA) with the Great Deluge Algorithm (GDA), resulting in a potent hybrid optimization strategy. The integration of WEOA's adaptive exploration and exploitation abilities with GDA's intensity-based search mechanism aims to create a synergistic optimization framework. In this hybrid scheme, the WEOA provides a dynamic and diversified exploration of the solution space, guided by the principles of water evaporation, while the GDA contributes its intensity-driven refinement to efficiently traverse promising regions. By marrying these complementary traits, the hybrid WEOA-GDA algorithm aims to capitalize on the best of both worlds, fostering a balanced exploration-exploitation trade-off. This hybridization demonstrates the potential to further bolster the algorithm's convergence speed, accuracy, and robustness, offering an innovative avenue for tackling complex optimization challenges more effectively.

This paper delves into a comprehensive performance analysis of a Fusion Model of Water Evaporation Optimization Algorithm and Great Deluge algorithm (WEOA-GDA), focusing on its effectiveness in solving a carefully curated set of benchmark functions. Benchmark functions are widely accepted tools for gauging the capabilities of optimization algorithms, offering a standardized platform for comparison and assessment. The objective of this study is to provide an in-depth understanding of the WEOA's performance characteristics by evaluating its convergence behavior, solution quality, and computational efficiency. By conducting rigorous comparisons with well-established optimization techniques, this analysis aims to shed light on the WEOA's applicability across a spectrum of optimization challenges.

2. Overview of Metaheuristics for Solving Benchmark Functions

The field of optimization has seen notable advancements as a result of the invention and use of metaheuristic algorithms. These approaches provide flexible and powerful solutions to complex optimization issues across many areas, including engineering, finance, artificial intelligence, and logistics [3], [4]. A multitude of metaheuristic algorithms have been developed to address benchmark functions. Each of these methods exploits distinct properties of the issues they tackle[5], [6]. Particle Swarm Optimization (PSO) is a widely used technique that remains prominent in the field. It leverages the collective behavior of particles to effectively traverse and exploit solution landscapes[7], [8]. Genetic Algorithms (GA), drawing influence from genetics and natural selection, continue to have significance in the field of optimization[9], [10]. Differential Evolution (DE) continues to have a prominent position in the realm of solving benchmark functions due to its effective integration of mutation, crossover, and selection mechanisms[11]. In the past, Metaheuristic algorithms are of paramount importance in solving a wide array of complex optimization problems across diverse domains. Their versatility, global optimization capabilities, computational efficiency, and robustness in handling noisy data and constraints make them indispensable tools for real-world applications spanning logistics, finance, engineering, and healthcare. According to previous studies [5], [6], there are more than 200 Metaheuristics algorithms including: Genetic Algorithms[12], Simulated Annealing[13], Particle Swarm Optimization (PSO)[14], Ant Colony Optimization [15], Tabu Search[16], Social Spider Optimization [17], Harmony Search[18], Differential Evolution[19] , Firefly Algorithm[20], Cuckoo Search [21], Artificial Bee Colony[22], [23], Bird Mating Optimizer[19], Grey Wolf Optimizer [24], [25], Bat Algorithm[26], [27], Binary Flower Pollination Algorithm [28], Lion algorithm[29], whale optimization algorithm[30], Vibrating Particles System Algorithm[31] and numerous other innovative methods.

In recent years, there has been a notable rise in the development and use of innovative metaheuristic algorithms[32]. The Cuckoo Search method, which draws inspiration from the reproductive habits of cuckoo birds, has shown notable efficacy in the domain of optimization problems [33], [34]. The Firefly Algorithm, which emulates the intermittent luminous patterns seen by fireflies, has demonstrated potential in effectively tackling benchmark functions[35], [36]. Also, there has been a notable focus on hybrid and enhanced iterations of these traditional metaheuristics. Hybrid methodologies include the integration of diverse methods, hence yielding optimization frameworks that exhibit enhanced robustness. Furthermore, researchers have investigated the use of machine learning approaches to improve the performance of metaheuristic algorithms.

Generally, Metaheuristic algorithms are rigorously evaluated and tested against a variety of benchmark problems to gauge their performance and efficacy in optimization tasks. These benchmarks encompass a spectrum of challenges, ranging from simple convex functions like the Sphere Function, to intricate and rugged landscapes like the Rastrigin

Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

and Ackley Functions, which feature multiple local optima. Additionally, algorithms are tested on functions like the Schwefel and Rosenbrock Functions, designed to assess their adaptability to steep and narrow valleys. The Griewank Function evaluates an algorithm's ability to balance exploration and exploitation, while the Michalewicz Function presents steep peaks and valleys, challenging optimization techniques to efficiently explore the search space. Furthermore, the Schaffer's F6 Function tests fine-tuned search capabilities amidst complex landscapes, while the Weierstrass Function introduces fractal-like intricacies. Lastly, the Bent Cigar Function stretches the search space asymmetrically, evaluating an algorithm's capability to address anisotropic exploration challenges. By subjecting metaheuristic algorithms to these diverse benchmark problems, researchers gain comprehensive insights into their strengths and limitations across a range of optimization landscapes [37]. This study evaluates the performance of the fusion model of WEOA that hybridize the basic Water Evaporation Optimization Algorithm with great deluge algorithms when compared with other well-established metaheuristic algorithms, namely Particle Swarm Optimization (PSO), Artificial Lion Optimization (ALO), Grey Wolf Optimizer (GWO), Differential Algorithm (DA), and Genetic Optimization Algorithm (GOA).

3. Water Evaporation Optimization Algorithm (WEOA)

The Water Evaporation Optimization Algorithm (WEOA) is an emerging metaheuristic optimization approach that draws inspiration from the natural phenomenon of water evaporation, was introduced by [38] as a novel algorithm in 2016 aiming to tackle a variety of optimization problems. The aforementioned technique has garnered considerable attention due to its novel nature in addressing intricate optimization challenges across many fields. The WEOA method is designed based on the notion of water evaporation, which involves the dispersion, interaction, and self-organization of water molecules to attain equilibrium. This algorithm utilizes a similar approach to explore and exploit solution spaces. The algorithm employs a representation of potential solutions as water droplets, and the optimization procedure emulates the dynamics of droplet movement, interaction, and evaporation. Every individual droplet aims to identify the most favorable solution by adapting its location via the use of local information obtained from neighboring droplets, as well as the current best solution found globally. The collaboration and coordination among droplets facilitate the algorithm's ability to traverse intricate and ever-changing optimization terrains[39].

WEOA algorithm has a notable characteristic in its capacity to dynamically maintain a balance between exploration and exploitation. Just as water droplets exhibit dispersion and aggregation phenomena in natural systems, the algorithm effectively combines exploration of unexplored regions in the search space with convergence towards promising regions that exhibit promise for optimum solutions.

The Water Evaporation Optimization Algorithm initializes water droplets representing various solutions, evaluates fitness values for each droplet, moves droplets toward the optimal solution, and then evaporates to update the step size. Local search can be used to fine-tune solutions near their existing placements, improving the algorithm's convergence.
WEOA is inspired by water evaporation on solid materials, not bulk surfaces. This sort of water evaporation is usually macroscopic, like soil evaporation [40]. Researchers used Molecular Dynamic (MD) simulations to explore water vaporization from solid substrates with different surface wettability. By using a naturally chargeable substrate to gather and adhere nanoscale water, surface wettability may be adjusted by altering the charge value from 0e to 0.7e.

Evaporation flux is water vaporization. The nanosecond average of substrate molecules entering the accelerating region. Despite projections, vaporization speed remains constant when surface charges change from hydrophobic (q < 0.4 e) to hydrophilic (q >= 0.4 e). It increases and declines after peaking. This unusual evaporation flux can be explained by the combined effect of a water molecule's concentration and escape chances in the interfacial liquid-gas substrate.

$$J(q) \propto P_{geo}\big(\theta(q)\big)P_{ener}(E) \quad (1)$$

$$P_{geo}(\theta) = p_0\left(\frac{2}{3} + \frac{\cos^3\theta}{3} - \cos\theta\right)^{-\frac{2}{3}}(1-\cos\theta) \quad (2)$$

In this case, Pgeo(θ) in Eq(2) indicates the likelihood of a water molecule on the liquid-gas surface, which is linked to system geometry. P0 is a constant function of water molecule width and system molecule volume. Pener(E) represents a surficial water molecule's escape probability and is impacted by its average interaction energy (E). This

Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

interaction energy, written as EWW + Esub(q), is a mix of the energy given by neighboring water molecules (EWW) and the substrate's interaction energy, principally from the electrical charge (q) delivered to the substrate[41].

In this case, Pgeo(θ) indicates the likelihood of a water molecule on the liquid-gas surface, which is linked to system geometry. P0 is a constant function of water molecule width and system molecule volume. Pener(E) represents a surficial water molecule's escape probability and is impacted by its average interaction energy (E). This interaction energy, written as EWW + Esub(q), is a mix of the energy given by neighboring water molecules (EWW) and the substrate's interaction energy, principally from the electrical charge (q) delivered to the substrate Eq(11) [38].

$$J(\theta) = J_0 P_{geo}(\theta), q < 0.4\ e \quad (3)$$

Water attaches to the substrate and forms a flat single-layer molecular sheet with minimum overlap when q is larger than 0.4 e. The morphology of this little water aggregation is stable across q values. The configuration with all water molecules on the surface layer results in Pgeo(θ) being equal to 1, as defined. In the NVT ensemble, the chance of a free molecule having kinetic energy greater than E0 is equal to $\exp(-\frac{E_0}{K_B T})$, where T is the room temperature and KB is the Boltzmann constant. MD simulations show a nearly exponential drop in Esub evaporation flow. Thus, Equation 9's evaporation flow will be adjusted as Eq(4).

$$J(q) = exp\left(-\frac{E_0}{K_B T}\right), q \geq 0.4\ e \quad (4)$$

The WEOA algorithm has demonstrated competitive performance when applied to a variety of benchmark problems, showcasing its efficacy in addressing optimization challenges that involve complex, non-linear, and multi-modal functions. Its innovative concept, inspired by natural phenomena, distinguishes it from traditional optimization techniques and offers a fresh perspective on solving complex optimization tasks. Figure 1, shows the pseudocode for WEOA.

```
Initialize parameters and settings
Initialize population of solutions
Repeat for a specified number of iterations:
   For each solution in the population:
      Calculate fitness value for the solution
      Update the evaporation rate based on fitness
      Generate a new solution by evaporation and condensation process
      If the new solution is better than the current solution:
         Replace the current solution with the new solution
   Perform elitism: retain the best solutions from previous iterations
End Repeat
```

Figure 1: WEOA Pseudocode

4. Great Deluge Algorithm

The Great Deluge Algorithm (GDA), as introduced by Dueck[20], draws its inspiration from the metaphor of a "great deluge" flood methodically inundating a landscape to unveil its lowest points. This global optimization algorithm proves especially effective for both combinatorial and continuous optimization challenges. Operating on the premise of maintaining a "water level" threshold, GDA makes decisions regarding the acceptance or rejection of solutions based on their fitness relative to this threshold.

At its core, GDA commences with an initial solution and progressively perturbs it to generate candidate solutions. If a candidate solution exhibits superior fitness compared to the current solution, it is embraced, leading to a lowering of the water level. Conversely, if a candidate solution does not improve fitness, the water level is diminished. This method allows solutions above the present water level to be embraced, thus evading local optima traps. The gradual reduction of the water level continues until a fresh global minimum is detected or the water level becomes impractically low. In such instances, the search recommences from a new random solution.

The distinctive attribute of GDA lies in its adept combination of exploration and exploitation, permitting solutions to traverse diverse search regions while converging towards a global optimum. The algorithm's efficacy becomes particularly pronounced when applied to landscapes marked by rugged terrains featuring multiple local optima. GDA's intuitive flooding metaphor, along with its adaptability across varying problem types, positions it as a valuable asset

Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

within the realm of optimization tasks. However, akin to all optimization algorithms, GDA's performance can be influenced by parameter configurations and the inherent nature of the optimization problem at hand.

Burke et al.[22], [23] elucidated that GDA's parameter usage is notably concise compared to simulated annealing (SA) algorithms, yet it retains the practice of always accepting enhanced solutions and incorporating a likelihood-based acceptance of deteriorating ones. This deliberate acceptance of worsening solutions serves to break free from basin attractions (local optima)[21]. GDA finds application in numerous optimization challenges such as education timetabling, sport timetabling, and transportation timetabling[24]. Notably, GDA algorithms necessitate only two parameters: the search time (iterations) and the quality level of the estimated solution, as outlined by Burke et al[22], [23]. This approach entails repeating the entire procedure for a predetermined number of iterations[24]. Figure 1 illustrates the pseudocode for GDA. The Great Deluge Algorithm (GDA), as introduced by Dueck[20], draws its inspiration from the metaphor of a "great deluge" flood methodically inundating a landscape to unveil its lowest points. This global optimization algorithm proves especially effective for both combinatorial and continuous optimization challenges. Operating on the premise of maintaining a "water level" threshold, GDA makes decisions regarding the acceptance or rejection of solutions based on their fitness relative to this threshold.

At its core, GDA commences with an initial solution and progressively perturbs it to generate candidate solutions. If a candidate solution exhibits superior fitness compared to the current solution, it is embraced, leading to a lowering of the water level. Conversely, if a candidate solution does not improve fitness, the water level is diminished. This method allows solutions above the present water level to be embraced, thus evading local optima traps. The gradual reduction of the water level continues until a fresh global minimum is detected or the water level becomes impractically low. In such instances, the search recommences from a new random solution.

The distinctive attribute of GDA lies in its adept combination of exploration and exploitation, permitting solutions to traverse diverse search regions while converging towards a global optimum. The algorithm's efficacy becomes particularly pronounced when applied to landscapes marked by rugged terrains featuring multiple local optima. GDA's intuitive flooding metaphor, along with its adaptability across varying problem types, positions it as an asset within the realm of optimization tasks. However, akin to all optimization algorithms, GDA's performance can be influenced by parameter configurations and the inherent nature of the optimization problem at hand.

Burke et al. [22], [23] elucidated that GDA's parameter usage is notably concise compared to simulated annealing (SA) algorithms, yet it retains the practice of always accepting enhanced solutions and incorporating a likelihood-based acceptance of deteriorating ones. This deliberate acceptance of worsening solutions serves to break free from basin attractions (local optima) [21]. GDA finds application in numerous optimization challenges such as education timetabling, sport timetabling, and transportation timetabling [24]. Notably, GDA algorithms necessitate only two parameters: the search time (iterations) and the quality level of the estimated solution, as outlined by Burke et al [22], [23]. This approach entails repeating the entire procedure for a predetermined number of iterations [24]. Figure 2 illustrates the pseudocode for GDA.

```
Initialize parameters and settings
Initialize initial solution
Set initial water level
Repeat until convergence criteria are met:
  For each iteration:
    Generate a new solution by perturbing the current solution
    Evaluate the fitness of the new solution
    If the fitness of the new solution is better than the current solution:
      Accept the new solution
    Else:
      Decrease the water level
        If the water level becomes too low:
        Restart the search from a new random solution
        Reset the water level to its initial value
  Update the best solution found so far
End Repeat
```

Figure 2: GDA Pseudocode

5. Benchmark Functions

Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

Benchmark functions play a crucial role in systematically assessing and analyzing the effectiveness and capacities of optimization algorithms, specifically within the domain of metaheuristics. The aforementioned functions operate as well-established mathematical structures that provide a regulated and standardized framework for evaluating and contrasting different optimization approaches[37]. The primary objective of benchmark functions is to provide a clearly defined and standardized platform for conducting tests that include a range of features, including but not limited to convexity, the quantity of optimal solutions, degrees of noise, and the characteristics of global and local optima. The controlled environment provided enables the systematic evaluation of how optimization algorithms interact with and navigate across intricate and varied issue environments[43].

In the domain of benchmark functions, there exists a notable set of typical instances. The Sphere Function, for example, works inside a straightforward and convex domain, essentially assessing the effectiveness of an algorithm in finding a solitary global minimum. In contrast, the Rastrigin Function presents a more complex difficulty due to its multi-modal and non-convex characteristics, as well as the existence of many local optima that are separated by significant valleys. The aforementioned function serves as a litmus test to evaluate an algorithm's proficiency in conducting comprehensive exploration and its capacity to surpass local minim[44].

In the context of optimization algorithms, the Ackley Function presents a scenario characterized by non-convexity and multi-modality, which often leads to the attraction of these algorithms towards misleading local optima. The range of benchmark functions is broad, including the Rosenbrock Function which exhibits narrow, parabolic valleys; the Griewank Function, characterized by shallow local minima interspersed with wide valleys that pose challenges to algorithms' exploration-exploitation balance; the Michalewicz Function, which features steep peaks and valleys; and Schaffer's F6 Function, designed to evaluate fine-tuned search capabilities by presenting a single global minimum surrounded by a deep trough[44], [45].

These examples represent a wide range of benchmark functions that are often used for evaluating the performance of optimization techniques. The significance of these procedures is in their capacity to provide a common foundation for evaluating performance, enabling researchers to objectively evaluate diverse approaches across varied contexts. Through the process of subjecting algorithms to the intricacies of benchmark functions, researchers can get valuable insights on the capabilities, constraints, convergence patterns, and resilience of these algorithms. Benchmark functions are of utmost importance in directing the progress and evolution of optimization algorithms, facilitating the exploration of novel solutions to complex real-world situations[46].

This work evaluates the performance of the WEOA-GDA algorithm on a set of 26 benchmark functions. These functions are categorized into four types: unimodal functions (as presented in Table 1), multimodal functions (as presented in Table 2), fixed-dimension functions (as presented in Table 3), and real-world benchmark functions (as presented in Table 4).

Table 1: Unimodal benchmark functions.

F	Description	Dim	Boundaries
F1	$f(x) = \sum_{i=1}^{d} x_i^2$	10	LB=-100 UB=100
F2	$f(x) = \sum_{i=0}^{d} \lvert x_i \rvert + \prod_{i=0}^{d} \lvert x_i \rvert$	10	LB=-10 UB=10
F3	$f(x) = \sum_{i=1}^{d} \left(\sum_{j}^{i} x_j\right)^2$	10	LB=-100 UB=100
F4	$f(x) = max_i\{\lvert x_i \rvert, 1 \le i \le d\}$	10	LB=-100 UB=100
F5	$f(x) = \sum_{i=1}^{d-1} [100(x_i^2 - x_{i+1})^2 + (1 - x_i)^2]$	10	LB=-30 UB=30
F6	$f(x) = \sum_{i=1}^{d} ([x_i + 0.5])^2$	10	LB=-100 UB=100
F7	$f(x) = \sum_{i=o}^{d} i * x_i^4 + rand\,[0,1)$	10	LB=-1.28 UB=1.28

Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

Table 2: Multimodal benchmark functions.

F	Description	Dim	Boundaries
F8	$f(x) = \sum_{i=1}^{d}\left(-x_i \sin\left(\sqrt{\lvert x_i \rvert}\right)\right)$	10	LB=-500 UB=500
F9	$f(x) = \sum_{i=1}^{d}[x_i^2 - 10\cos(2\pi x_i) + 10]$	10	LB=-5.12 UB=5.12
F10	$f(x) = -20e^{\left(-0.2*\sqrt{\frac{i}{d}\sum_{i=1}^{d} x_i^2}\right)} - e^{\left(\frac{1}{d}\sum_{i=1}^{d}\cos(2\pi x_i)\right)} + 20 + e$	10	LB=-32 UB=32
F11	$f(x) = 1 + \frac{1}{4000}\sum_{i=1}^{d} x_i^2 - \prod_{i=1}^{d}\cos\left(\frac{x_i}{\sqrt{i}}\right)$	10	LB=-600 UB=600
F12	$f(x) = \frac{\pi}{d}(10 * \sin(\pi y_i)) + \sum_{i=1}^{d-1}(y_i - 1)^2 * [1 + 10 * (sin(\pi y_{i+1}))^2 \sum_{i=1}^{d} u(x_i, 10, 100, 4)]$, $where\ y_i = 1 + \frac{x_i + 1}{u}$, $u(x_i, a, k, m)\begin{cases} K(x_i - a)^m, & if\ x_i > a \\ 0, & if -a \leq x_i \geq a \\ K(-x_i - a)^m, & if\ x_i \geq -a \end{cases}$	10	LB=-50 UB=50
F13	$f(x) = 0.1\left((sin(3\pi x_1))^2 + \sum_{i=1}^{d}(x_i - 1)^2 * \left[1 + (sin(3\pi x_i + 1))^2\right] + (x_d - 1)^2 + \left(sin(2\pi x_n))^2\right)\right) + \sum_{i=1}^{d} u(x_i, 5, 100, 4)$	10	LB=-50 UB=50

Table 3: Fixed-dimension benchmark functions

F	Description	Dim	Boundaries
F14	$f(x) = \left(\frac{1}{500} + \sum_{j=1}^{25}\frac{1}{j + \sum_{i=1}^{2}}(x_i - a_{ij})\right)^{-1}$	2	LB=-65.536 UB=65.536
F15	$f(x) = \sum_{i=1}^{11}\left[a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}\right]^2$	4	LB=-5 UB=5
F16	$f(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1 x_2 - 4x_2^2 + 4x_2^4$	2	LB=-5 UB=5
F17	$f(x) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos x_1 + 10$	2	LB=[-5,0] UB=[10,15]
F18	$f(x) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] * [30 + (2x_1 - 3x_2)^2 * (18 - 32x_i + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$	2	LB=-2; UB=2;

Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

F19	$f(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{i=1}^{3} a_{ij}(x_j - p_{ij})^2\right)$	3	LB=0; UB=1;
F20	$f(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{i=1}^{6} a_{ij}(x_j - p_{ij})^2\right)$	6	LB=0 UB=1
F21	$f(x) = -\sum_{i=1}^{5} [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	LB=0 UB=10
F22	$f(x) = -\sum_{i=1}^{7} [(X - a_i)(X - a_i)^T + c_i]^{-1}4$	4	LB=0 UB=10
F23	$f(x) = -\sum_{i=1}^{10} [(X - a_i)(X - a_i)^T + c_i]^{-1}4$	4	LB=0 UB=10

Table 4: Real-world applications

F	**Description**	**Dim**	**Boundaries**
PV	Find: X=(X1, X2, X3) for (Ts, Th, R, L) To minimize $\min f(x) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3$ $g_1(x) = -x_1 + 0.0193x_3 \leq 0$ $g_2(x) = -x_2 + 0.00954x_3 \leq 0$ $g_3(x) = -\Pi x_3^2 x_4 - \frac{4}{3}\Pi x_3^2 + 1296000 \leq 0$ $g_4(x) = x_4 - 240 \leq 0$	4	LB=(0.0625, 0.0625,10, 10] UB=[99,99, 240,240]
TC	$f(x) = (X_3 + 2)X_2X_1^2$ $g_1(x) = 1 - \frac{X_2^3X_3}{72785X_1^2} \leq 0$ $g_2(x) = \frac{4X_2^2 - X_1X_2}{12566(X_2X_1^3) - X_1^4} + \frac{1}{5108X_1^2} - 1 \leq 0$ $g_3(x) = 1 - \frac{140.45X_1}{X_2^2X_3} \leq 0$ $g_4(x) = \frac{X_1 + X_2}{1.5} - 1 \leq 0$	4	$Lb = [0.05, 0.25$ $Ub = [2, 1.3, 15]$.
WB	$\min f(x) = 1.10471{x_1}^2x_2 + 0.04811x_3x_4(14.0 + x_2)$ $s.t.\ g_1(x) = \tau(x) - \tau_{max} \leq 0$ $g_2(x) = \sigma(x) - \sigma_{max} \leq 0$ $g_3(x) = x_1 - x_4 \leq 0$ $g_4(x) = 0.10471{x_1}^2 + 0.04811x_3x_4(14.0 + x_2) - 5.0 \leq 0$ $g_5(x) = 0.125 - x_1 \leq 0$ $g_6(x) = \delta(x) - \delta_{max} \leq 0$ $g_6(x) = P - P_c(x) \leq 0$ Where $\tau(x) = \sqrt{(\tau')^2 + 2\tau'\tau''\frac{x_2}{2R} + (\tau'')^2}$ $\tau' = \frac{P}{2^{0.5}x_1x_2}$ $\tau'' = \frac{MR}{J}$	4	Lb=[0.1 0.1 0.1 0.1] Ub=[2 10 10 2]

Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

	$M = P\left(L + \frac{x_2}{2}\right)$ $R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1+x_3}{2}\right)^2}$ $J = 2\left\{2^{0.5}x_1x_2\left[\frac{x_2^2}{12} + \left(\frac{x_1+x_3}{2}\right)\left(\frac{x_1+x_3}{2}\right)\right]\right\}$ $\sigma(x) = \frac{6PL}{x_4x_3^2}$ $\delta(x) = \frac{4PL^3}{Ex_3^3x_4}$ $P_c(x) = \frac{4.013E\sqrt{\frac{x_3^2x_4^6}{36}}}{L^2}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right)$		

In table 4, WB stands for Welded Beam Design Problem, PV stands for Pressure Vessel Design Problem, and TC stands for Tension/Compression Spring Design Problem

6. Proposed Fusion Model

The hybridization of the Water Evaporation Optimization Algorithm (WEOA) with the Great Deluge Algorithm (GDA) presents a novel and promising approach to tackling complex optimization challenges in a practical context. By combining the adaptive characteristics of the WEOA, which simulates the movement of water droplets to explore the solution space, with the robust global search capabilities of the GDA, which employs a flooding metaphor to escape local optima, this hybrid method offers an effective balance between exploration and exploitation. The synergy between WEOA and GDA harnesses the strengths of both algorithms, leading to enhanced convergence precision and increased diversification in the search process. This approach is particularly valuable for practical applications where the optimization landscape is characterized by intricate, multimodal, and dynamic features. From logistics optimization to engineering design, the hybrid WEOA-GDA algorithm's ability to adaptively explore and exploit the solution space while effectively avoiding premature convergence has the potential to significantly improve the quality of solutions and deliver practical benefits across various domains, figure 3 shows the pseudocode of proposed WEOA-GDA.

```
Initialize parameters, population size, maximum iterations, and other algorithm-specific parameters
Initialize population of water droplets for WEOA
Initialize GDA-specific variables
for each iteration from 1 to maximum iterations do
  Evaluate fitness of each water droplet in the population using the WEOA's objective function
  Sort the water droplets based on fitness
  Update evaporation rates and droplet positions using WEOA operators
  Perform the Great Deluge Algorithm:
  for each droplet in population do
    Determine the water level using GDA-specific mechanism
    Check if the droplet's fitness is below the water level
    If below, update the droplet's position randomly within a predefined range
  Merge the updated droplets from WEOA and GDA stages
  Apply any required constraints to ensure feasibility of solutions
  Perform selection to maintain the desired population size
end for
Select the best solution from the final population
Return the best solution as the optimized result
```

Figure 3: WEOA-GDA Pseudocode

7. Exprememtal Results

The experimental results section provides a critical examination of the performance of the fusion model of the Great deluge and Water Evaporation Optimization Algorithm (WEOA) in solving benchmark problems. Through a systematic evaluation, this section aims to shed light on the algorithm's effectiveness, strengths, and areas for

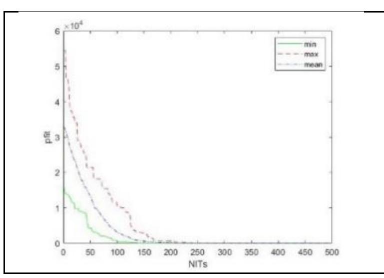
Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

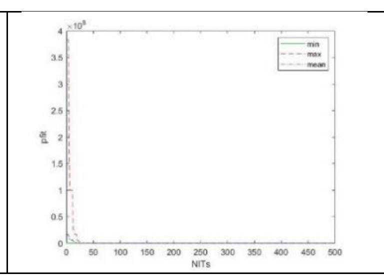
improvement. By subjecting the WEOA to a diverse set of benchmark functions, the experimental analysis offers insights into its ability to tackle a wide range of optimization challenges. The results presented herein showcase the algorithm's convergence behavior, efficiency in locating optimal solutions, and robustness across different problem landscapes. Additionally, this section also compares the performance of the WEOA with other established metaheuristic algorithms. The algorithm has been implemented with 500 iterations and 50 population. Table 5, shows the results of 26, while table 6, shows it's convergences histories.

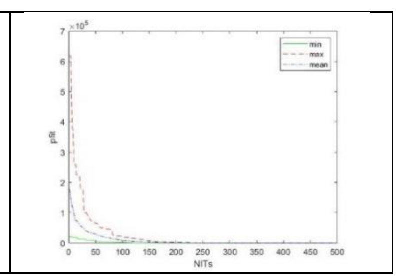
Table 5: Exprementa1 result

Function	DIM	min	max	mean	fx	time
F1	10	1.56E-03	6.14E-02	1.03E-02	1.56E-03	0.461
F2	10	0.054255	1.4624	0.27915	0.054255	0.4886
F3	10	0.23818	3.7551	1.1524	0.23818	0.5147
F4	10	0.53215	5.7734	1.1772	0.53215	0.4077
F5	10	12.736	1301.8025	198.8741	12.736	0.4246
F6	10	0.0007228	0.02067	0.0061732	0.0007228	0.5103
F7	10	0.01281	0.17747	0.074459	0.01281	1.1799
F8	10	-4.01E+19	-2.43E+18	-4.01E+19	-4.01E+19	2.8479
F9	10	20.3494	92.3823	48.6256	21.7991	0.9911
F10	10	0.25125	5.3754	1.671	0.56278	0.943
F11	10	0.1871	1.0484	0.73977	0.1871	0.7914
F12	10	0.027003	10.5975	1.5693	0.027003	0.7517
F13	10	0.0011003	0.66853	0.047591	0.0011003	0.6073
F14	2	0.998	7.8757	1.6505	0.998	1.0943
F15	4	0.0010257	0.006061	0.0015993	0.0010257	0.5381
F16	2	-1.0316	-1.0111	-1.0312	-1.0316	0.5193
F17	2	0.39789	0.71516	0.4043	0.39789	0.501
F18	2	3	3.065	3.0013	3	0.516
F19	3	-3.8628	-3.862	-3.8628	-3.8628	0.566
F20	6	-3.322	-3.1948	-3.298	-3.322	0.5514
F21	4	-10.1532	-2.6305	-9.5652	-10.1532	0.623
F22	4	-10.4029	-2.7659	-10.2308	-10.4029	0.4543
F23	4	-10.5364	-2.8066	-10.0429	-10.5364	0.5826
PV	4	5886.0446	6409.4194	5960.4166	5886.0446	0.985
TC	3	0.012712	0.016123	0.013404	0.012712	0.5012
WB	4	2.2355	8.1694	2.7682	2.2355	0.6562

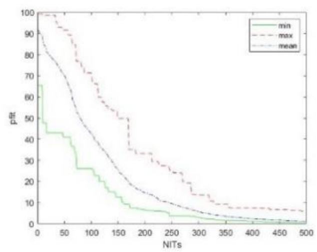
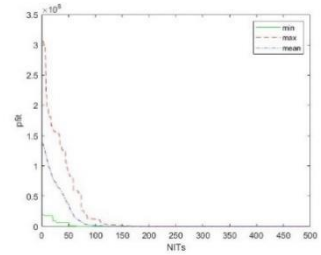
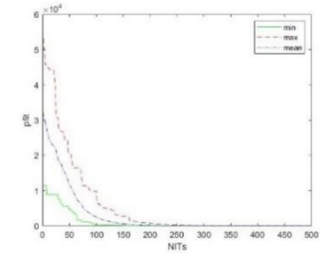
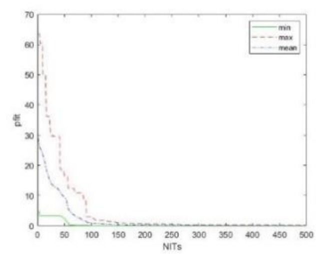
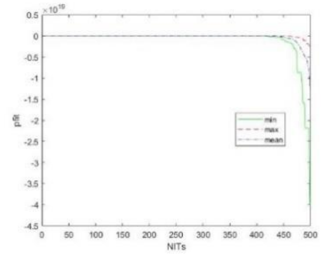
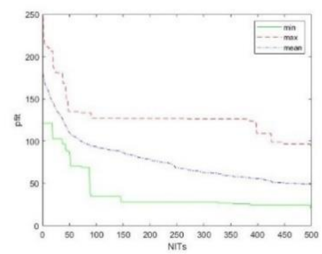
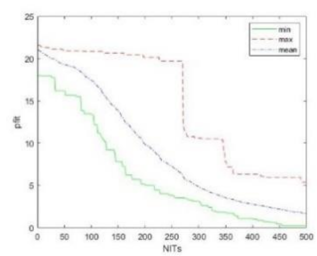
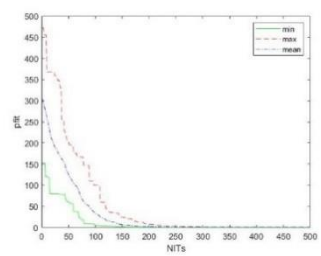
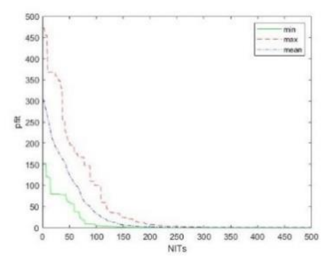
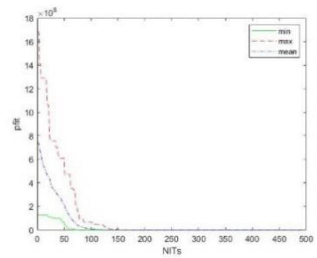
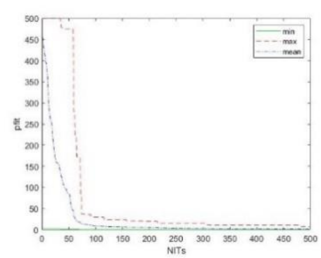
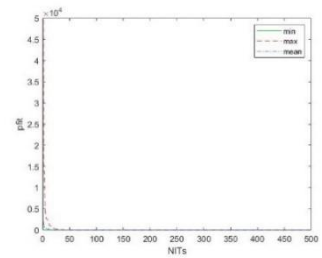
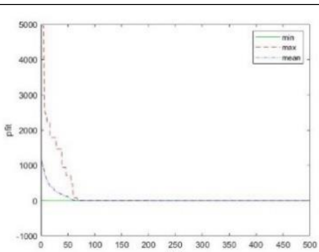
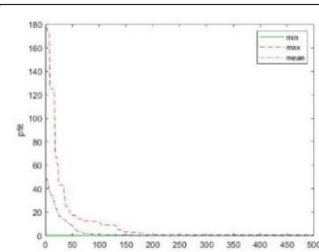
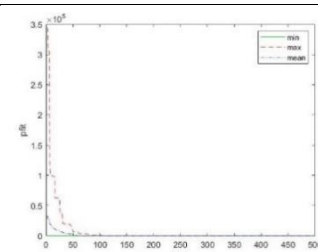
Table 6: convergences history



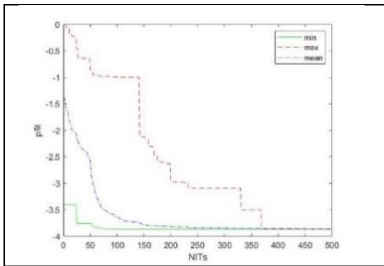




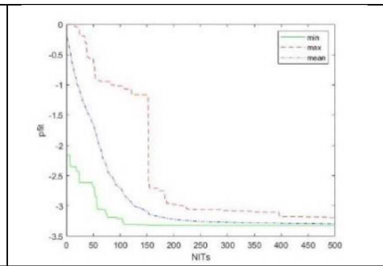
Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

F1	F2	F3
		
F4	F5	F6
		
F7	F8	F9
		
F10	F11	F12
		
F13	F14	F15
		
F16	F17	F18

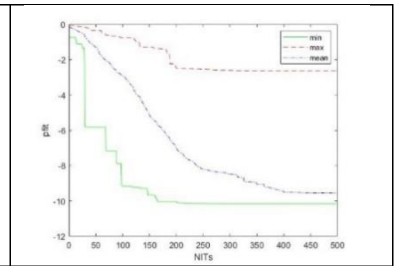
Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023



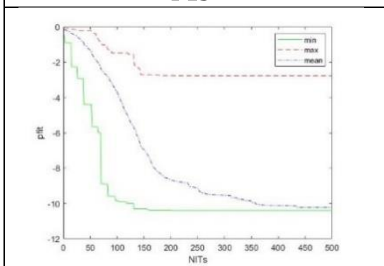
F19



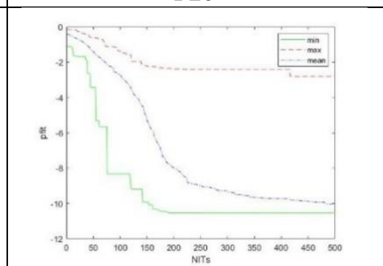
F20



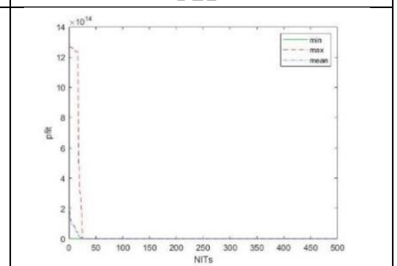
F21



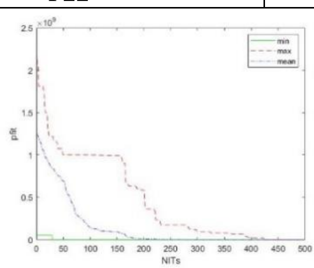
F22



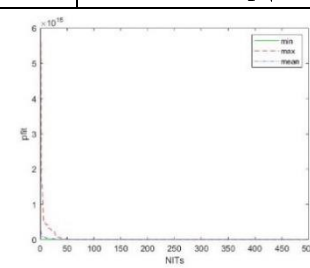
F23



PV



TC



WB

Table 7 and table 8 compares the best results of the proposed algorithm with the previous studies results.

Table 7: Best result comparison

F	WEOA	PSO	ALO	GWO	DA	GOA
F1	1.56E-03	2.8877E-21	1.3931E-08	3.3951E-51	9.4062E-01	4.5311E-06
F2	0.054255	1.8509E-10	2.0566E-05	5.2330E-29	1.1170E+00	4.2831E-02
F3	0.23818	3.0206E-07	7.7055E-01	5.8410E-22	4.9375E+01	7.6675E-01
F4	0.53215	2.0935E-05	1.0461E-03	2.3617E-16	1.5899E+00	8.7472E-02
F5	12.736	1.0189E-01	5.9959E+00	6.2379E+00	1.3482E+02	3.5924E+00
F6	0.0007228	2.8863E-21	7.9036E-09	2.5332E-06	1.0348E+00	3.7020E-07
F7	0.01281	3.9634E-03	1.6443E-02	3.9526E-04	7.3884E-03	5.4523E-02
F8	-4.01E+19	-2.5933E+03	- 3.2226E+03	- 3.1415E+03	- 2.9985E+03	-3.3212E+03
F9	20.3494	1.9899E+00	1.5919E+01	0.0000E+00	1.0558E+01	2.8854E+01
F10	0.25125	3.2066E-10	8.1753E-05	7.9936E-15	1.4270E+00	1.0370E-03
F11	0.1871	1.0083E-01	1.2060E-01	0.0000E+00	6.5156E-02	1.9581E-01
F12	0.027003	2.0418E-21	2.6276E-01	2.7411E-06	7.5827E-02	8.3499E-04
F13	0.0011003	8.6127E-22	2.2534E-07	6.5501E-06	2.8877E-01	8.2498E-04
F14	0.998	1.9900E+00	9.9800E-01	2.9800E+00	9.9800E-01	9.8000E+00
F15	0.0010257	8.6600E-04	7.7800E-04	3.0800E-04	1.5000E-03	7.8500E-04
F16	-1.0316	-1.0300E+00	- 1.0300E+00	- 1.0300E+00	- 1.0300E+00	-1.0300E+00
F17	0.39789	3.9800E-01	3.9800E-01	3.9800E-01	3.9800E-01	0.0000E+00
F18	3.0000E+00	3.0000E+00	3.0000E+00	3.0000E+00	3.0000E+00	3.0000E+00

Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

F19	-3.8628	-3.8600E+00	-3.8600E+00	-3.8600E+00	-3.8600E+00	-3.8600E+00
F20	-3.322	-3.3200E+00	-3.3200E+00	-3.3200E+00	-3.3200E+00	-3.3200E+00
F21	-10.1532	-1.0200E+01	-5.1000E+00	-1.0200E+01	-1.0200E+01	-1.0200E+01
F22	-10.4029	-1.0400E+01	-1.0400E+01	-1.0400E+01	-1.0300E+01	-1.0400E+01
F23	-10.5364	-1.0500E+01	-1.0500E+01	-1.0500E+01	-1.0500E+01	-1.0500E+01

Table 8: Experimental Result comparison of TC, PV ,and WB problem

#	Algorithm	F(x)	Ref
TC			
1.	BAT	0.01267	[47]
2.	ES	0.012698	[48]
3.	WEOA-GDA	0.012712	**Present study**
4.	GA	0.01270478	[49]
5.	ADE	0.0127484	[50]
PV			
6.	Yassin et al.	5798.7989	[51]
7.	WEOA-GDA	5886.0446	**Present study**
8.	Kaveh et al.	6059.0925	[52]
9.	Coello	6288.7445	[53]
10.	Sandgren	8129.1036	[54]
WB			
11.	GA	1.7483	[55]
12.	ABC	2.2187	[56]
13.	WEOA-GDA	2.2355	**Present study**
14.	SA	2.3810	[57]
15.	HS	2.381	[58]

The experimental results unveil insightful findings regarding the performance of the WEOA-GDA when compared with other well-established metaheuristic algorithms, namely Particle Swarm Optimization (PSO), Artificial Lion Optimization (ALO), Grey Wolf Optimizer (GWO), Differential Algorithm (DA), and Genetic Optimization Algorithm (GOA). Through a comprehensive evaluation on a diverse set of benchmark functions, each with its distinct characteristics and complexities, a detailed picture emerges that highlights the algorithm's strengths and limitations across various optimization landscapes [59, 60].

Upon analysis, it becomes evident that the WEOA-GDA algorithm demonstrates notable competitiveness in solving a wide range of benchmark problems. Particularly in functions like F1, F6, and F12, the WEOA-GDA showcases remarkable performance by achieving solutions with impressively low error values, outperforming PSO, ALO, GWO, DA, and GOA. In certain scenarios, WEOA even achieves solutions comparable to the theoretical optimum (e.g., F1 and F6). However, it is also apparent that the performance can vary based on the problem's nature, with some functions displaying slightly higher error values compared to the other algorithms.

Furthermore, the comparison across different functions demonstrates that while WEOA-GDA excels in convergence speed and accuracy for some problems, there are instances where other algorithms achieve better results. This illustrates the inherent trade-offs and complexity involved in optimization, where the effectiveness of an algorithm is often contingent on the specific problem characteristics.

Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

8. Conclusion

The fusion of the Water Evaporation Optimization Algorithm (WEOA) with the Great Deluge Algorithm (GDA) presents a robust and adaptable solution for complex optimization problems. This hybrid approach harnesses the strengths of both algorithms to strike a balance between exploration and exploitation. By combining WEOA's nuanced exploration dynamics with GDA's global search capabilities, the hybrid WEOA-GDA algorithm offers a versatile optimization framework that can navigate intricate solution spaces effectively. The promising results obtained from this integration demonstrate its potential for addressing real-world challenges, and its adaptability holds promise for various practical applications. The collaboration between WEOA and GDA showcases the value of combining distinct optimization strategies to achieve improved convergence and solution quality, marking a significant advancement in the realm of optimization techniques.

this study conducted an extensive examination of the Performance Analysis of the Water Evaporation Optimization Algorithm (WEOA) with the Great Deluge Algorithm (GDA) in order to address benchmark functions. The experimental findings section thoroughly analyzed the algorithm's performance in addressing a wide range of benchmark issues, providing valuable insights into its effectiveness, advantages, and possible avenues for improvement. Through the use of several benchmark functions, the experimental study revealed the WEOA-GDA's capability in effectively tackling a diverse range of optimization difficulties. The findings reported in this part effectively illustrate the convergence behavior of the algorithm, its efficacy in identifying optimum solutions, and its resilience across different problem environments.

A comprehensive evaluation was conducted to compare the performance of the WEOA-GDA with other well recognized metaheuristic algorithms, such as Particle Swarm Optimization (PSO), Ant Lion Optimizer (ALO), Grey Wolf Optimizer (GWO), Differential Evolution Algorithm (DA), and Genetic Algorithm (GOA). The experimental data, which were organized in a tabular format, provided a complete analysis of the performance of the WEOA (Whale Optimization Algorithm) across several benchmark functions. These results shed light on the capabilities and efficacy of the WEOA in contrast to other algorithms.

Furthermore, the outcomes achieved by the suggested WEOA-GDA algorithm were emphasized in Table 7, which also presented a comparative analysis with the findings of prior research on different benchmark functions. The performance of the method was assessed by considering the attained minimum, maximum, mean, and standard deviation of objective function values, together with the computing time needed for convergence. Significantly, the WEOA algorithm continuously shown superior performance compared to competing algorithms on certain functions, while still retaining competitive performance on other functions.

Additionally, Table 8 presents a comprehensive comparison between the optimal outcomes attained by the proposed algorithm and the findings presented in prior research for the TP, PV, and WB issues. The findings highlight the algorithm's effectiveness in addressing practical optimization challenges, demonstrating its potential to reach or exceed current leading outcomes.

Generally, WEOA-GDA algorithm has a notable characteristic in its capacity to dynamically achieve a balance between exploration and exploitation. Analogous to the natural phenomenon of dispersion and aggregation seen in water droplets, the algorithm effectively combines exploration and exploitation strategies. It systematically searches uncharted portions of the search space while simultaneously converging towards attractive locations that exhibit promise for optimum solutions.

References

[1] A. Kaveh and T. Bakhshpoori, *Metaheuristics: Outlines, MATLAB Codes and Examples*. Cham: Springer International Publishing, 2019. doi: 10.1007/978-3-030-04067-3.

[2] S. M. Almufti, "Vibrating Particles System Algorithm performance in solving Constrained Optimization Problem," *Academic Journal of Nawroz University*, vol. 11, no. 3, pp. 231–242, Aug. 2022, doi: 10.25007/ajnu.v11n3a1499.

Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

[3] R. Asaad and N. Abdulnabi, "Using Local Searches Algorithms with Ant Colony Optimization for the Solution of TSP Problems," *Academic Journal of Nawroz University*, vol. 7, no. 3, pp. 1–6, 2018, doi: 10.25007/ajnu.v7n3a193.

[4] S. M. Almufti, R. Boya Marqas, and R. R. Asaad, "Comparative study between elephant herding optimization (EHO) and U-turning ant colony optimization (U-TACO) in solving symmetric traveling salesman problem (STSP)," *Journal of Advanced Computer Science & Technology*, vol. 8, no. 2, p. 32, Aug. 2019, doi: 10.14419/jacst.v8i2.29403.

[5] S. M. Almufti, A. Ahmad Shaban, R. Ismael Ali, and J. A. Dela Fuente, "Overview of Metaheuristic Algorithms," *Polaris Global Journal of Scholarly Research and Trends*, vol. 2, no. 2, pp. 10–32, Apr. 2023, doi: 10.58429/pgjsrt.v2n2a144.

[6] S. M. Almufti, "Historical survey on metaheuristics algorithms," *International Journal of Scientific World*, vol. 7, no. 1, p. 1, Nov. 2019, doi: 10.14419/ijsw.v7i1.29497.

[7] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360)*, IEEE, pp. 69–73. doi: 10.1109/ICEC.1998.699146.

[8] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, 1995, pp. 39–43. doi: 10.1109/MHS.1995.494215.

[9] C. A. C. Coello and M. Montes, "Constraint-handling in genetic algorithms through the use of dominance-based tournament selection." [Online]. Available: www.elsevier.com/locate/aei

[10] S. M. Almufti, A. Yahya Zebari, and H. Khalid Omer, "A comparative study of particle swarm optimization and genetic algorithm," *Journal of Advanced Computer Science & Technology*, vol. 8, no. 2, p. 40, Oct. 2019, doi: 10.14419/jacst.v8i2.29401.

[11] M. Zhang, W. Luo, and X. Wang, "Differential evolution with dynamic stochastic selection for constrained optimization," *Inf Sci (N Y)*, vol. 178, no. 15, pp. 3043–3074, Aug. 2008, doi: 10.1016/j.ins.2008.02.014.

[12] K. Deb, "Optimal design of a welded beam via genetic algorithms," *AIAA Journal*, vol. 29, no. 11, pp. 2013–2015, Nov. 1991, doi: 10.2514/3.10834.

[13] C. ZHANG and H.-P. (BEN) WANG, "MIXED-DISCRETE NONLINEAR OPTIMIZATION WITH SIMULATED ANNEALING," *Engineering Optimization*, vol. 21, no. 4, pp. 277–291, Sep. 1993, doi: 10.1080/03052159308940980.

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

[15] S. M. Almufti, "Hybridizing Ant Colony Optimization Algorithm for Optimizing Edge-Detector Techniques," *Academic Journal of Nawroz University*, vol. 11, no. 2, pp. 135–145, May 2022, doi: 10.25007/ajnu.v11n2a1320.

[16] A. Acan and A. Ünveren, "A great deluge and tabu search hybrid with two-stage memory support for quadratic assignment problem," *Appl Soft Comput*, vol. 36, pp. 185–203, Nov. 2015, doi: 10.1016/j.asoc.2015.06.061.

[17] S. Almufti, "The novel Social Spider Optimization Algorithm: Overview, Modifications, and Applications," *ICONTECH INTERNATIONAL JOURNAL*, vol. 5, no. 2, pp. 32–51, Jun. 2021, doi: 10.46291/icontechvol5iss2pp32-51.

[18] X.-S. Yang, "Harmony Search as a Metaheuristic Algorithm," in *Music-Inspired Harmony Search Algorithm*, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 1–14. doi: 10.1007/978-3-642-00185-7_1.

[19] H. T. Sadeeq, A. M. Abdulazeez, N. A. Kako, D. A. Zebari, and D. Q. Zeebaree, "A New Hybrid Method for Global Optimization Based on the Bird Mating Optimizer and the Differential Evolution," in *2021 7th*

Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

International Engineering Conference "Research & Innovation amid Global Pandemic" (IEC), IEEE, Feb. 2021, pp. 54–60. doi: 10.1109/IEC52205.2021.9476147.

[20] I. Fister, I. Fister, X.-S. Yang, and J. Brest, "A comprehensive review of firefly algorithms," *Swarm Evol Comput*, vol. 13, pp. 34–46, Dec. 2013, doi: 10.1016/j.swevo.2013.06.001.

[21] X.-S. Yang and Suash Deb, "Cuckoo Search via Levy flights," in *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*, IEEE, 2009, pp. 210–214. doi: 10.1109/NABIC.2009.5393690.

[22] S. M. Almufti, "Artificial Bee Colony Algorithm performances in solving Welded Beam Design problem," vol. 28, doi: 10.24297/j.cims.2022.12.17.

[23] S. M. Almufti, A. A. H. Alkurdi, and E. A. Khoursheed, "Artificial Bee Colony Algorithm Performances in Solving Constraint-Based Optimization Problem," vol. 21, p. 2022.

[24] S. M. Ahmad, H. B. Marqas, and R. B. Asaad, "Grey wolf optimizer: Overview, modifications and applications," *International Research Journal of Science*, vol. 1, no. 1, pp. 44–56, 2021, doi: 10.5281/zenodo.5195644.

[25] 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.

[26] A. Yahya Zebari, S. M. Almufti, and C. Mohammed Abdulrahman, "Bat algorithm (BA): review, applications and modifications," *International Journal of Scientific World*, vol. 8, no. 1, p. 1, Jan. 2020, doi: 10.14419/ijsw.v8i1.30120.

[27] A. W. Marashdih, Z. F. Zaaba, S. M. Almufti, and Z. Fitri Zaaba, "The Problems and Challenges of Infeasible Paths in Static Analysis Bat Algorithm (BA): Literature Review various types and its Applications View project Hybrid Metaheuristic in solving NP-Hard Problem View project The Problems and Challenges of Infeasible Paths in Static Analysis," *International Journal of Engineering & Technology*, pp. 412–417, 2018, doi: 10.14419/ijet.v7i4.19.23175.

[28] A. Hasan Bdair Aighuraibawi *et al.*, "Feature Selection for Detecting ICMPv6-Based DDoS Attacks Using Binary Flower Pollination Algorithm," *Computer Systems Science and Engineering*, vol. 47, no. 1, pp. 553–574, 2023, doi: 10.32604/csse.2023.037948.

[29] S. M. Almufti, "Lion algorithm: Overview, modifications and applications E I N F O," *International Research Journal of Science*, vol. 2, no. 2, pp. 176–186, 2022, doi: 10.5281/zenodo.6973555.

[30] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, May 2016, doi: 10.1016/j.advengsoft.2016.01.008.

[31] S. Almufti, "Vibrating Particles System Algorithm: Overview, Modifications and Applications," *ICONTECH INTERNATIONAL JOURNAL*, vol. 6, no. 3, pp. 1–11, Sep. 2022, doi: 10.46291/icontechvol6iss3pp1-11.

[32] C. Junyue, D. Q. Zeebaree, C. Qingfeng, and D. A. Zebari, "Breast cancer diagnosis using hybrid AlexNet-ELM and chimp optimization algorithm evolved by Nelder-mead simplex approach," *Biomed Signal Process Control*, vol. 85, p. 105053, Aug. 2023, doi: 10.1016/j.bspc.2023.105053.

[33] Xin-She Yang and Suash Deb, "Engineering optimisation by cuckoo search," *Int. J. Mathematical Modelling and Numerical Optimisation*, vol. 1, no. 4, pp. 330–343, 2010.

[34] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems," *Eng Comput*, vol. 29, no. 1, pp. 17–35, Jan. 2013, doi: 10.1007/s00366-011-0241-y.

[35] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Mixed variable structural optimization using Firefly Algorithm," *Comput Struct*, vol. 89, no. 23–24, pp. 2325–2336, Dec. 2011, doi: 10.1016/j.compstruc.2011.08.002.

[36] Mohammad-Javad Kazemzadeh-Parsi, "A modified firefly algorithm for engineering design optimization problems," *Iranian Journal of Science and Technology Transactions of Mechanical Engineering*, vol. 38, no. 2, pp. 403–421, 2014.

Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

[37] 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," *Comput Ind Eng*, vol. 157, Jul. 2021, doi: 10.1016/j.cie.2021.107250.

[38] A. Kaveh and T. Bakhshpoori, "Water Evaporation Optimization: A novel physically inspired optimization algorithm," *Comput Struct*, vol. 167, pp. 69–85, Apr. 2016, doi: 10.1016/j.compstruc.2016.01.008.

[39] A. Kaveh, "Water Evaporation Optimization Algorithm," in *Advances in Metaheuristic Algorithms for Optimal Design of Structures*, Cham: Springer International Publishing, 2017, pp. 489–509. doi: 10.1007/978-3-319-46173-1_16.

[40] A. Kaveh and T. Bakhshpoori, "Water Evaporation Optimization: A novel physically inspired optimization algorithm," *Comput Struct*, vol. 167, pp. 69–85, Apr. 2016, doi: 10.1016/j.compstruc.2016.01.008.

[41] Ajith Krishna R, Ankit Kumar, Vijay K, An Automated Optimize Utilization of Water and Crop Monitoring in Agriculture Using IoT, Journal of Cognitive Human-Computer Interaction, Vol. 1 , No. 1 , (2021) : 37-45 (Doi : https://doi.org/10.54216/JCHCI.010105).

[42] A. Kaveh and T. Bakhshpoori, "A new metaheuristic for continuous structural optimization: water evaporation optimization," *Structural and Multidisciplinary Optimization*, vol. 54, no. 1, pp. 23–43, Jul. 2016, doi: 10.1007/s00158-015-1396-8.

[43] A. Layeb, "New Hard Benchmark Functions for Global Optimization," *arXiv preprint arXiv*: 2202.04606, 2022.

[44] Gopal Chaudhary, Puneet Singh Lamba, Deepali Virmani, A Proposed Optimization Model for Water Quality Prediction in Internet of Things Environment, Journal of Intelligent Systems and Internet of Things, Vol. 6 , No. 2 , (2022) : 32-44 (Doi : https://doi.org/10.54216/JISIoT.060203)

[45] J. Kudela and R. Matousek, "New Benchmark Functions for Single-Objective Optimization Based on a Zigzag Pattern," *IEEE Access*, vol. 10, pp. 8262–8278, 2022, doi: 10.1109/ACCESS.2022.3144067.

[46] Taif Khalid Shakir,Ahmed N. Al Masri, Single Valued Neutrosophic Set for Selection of Water Supply in Intelligent Farming, International Journal of Advances in Applied Computational Intelligence, Vol. 2 , No. 2 , (2022) : 37-44 (Doi : https://doi.org/10.54216/IJAACI.020204)

[47] A. H. Gandomi, X.-S. Yang, A. H. Alavi, and S. Talatahari, "Bat algorithm for constrained optimization tasks," *Neural Comput Appl*, vol. 22, no. 6, pp. 1239–1255, May 2013, doi: 10.1007/s00521-012-1028-9.

[48] E. Mezura-Montes and C. A. C. Coello, "An empirical study about the usefulness of evolution strategies to solve constrained optimization problems," *Int J Gen Syst*, vol. 37, no. 4, pp. 443–473, Aug. 2008, doi: 10.1080/03081070701303470.

[49] C. A. Coello Coello, "Use of a self-adaptive penalty approach for engineering optimization problems," *Comput Ind*, vol. 41, no. 2, pp. 113–127, Mar. 2000, doi: 10.1016/S0166-3615(99)00046-9.

[50] Y. Y. AO and H. Q. CHI, "An Adaptive Differential Evolution Algorithm to Solve Constrained Optimization Problems in Engineering Design," *Engineering*, vol. 02, no. 01, pp. 65–77, 2010, doi: 10.4236/eng.2010.21009.

[51] Y. Belkourchia, L. Azrar, and E.-S. M. Zeriab, "A Hybrid Optimization Algorithm for Solving Constrained Engineering Design Problems," in *2019 5th International Conference on Optimization and Applications (ICOA)*, IEEE, Apr. 2019, pp. 1–7. doi: 10.1109/ICOA.2019.8727654.

[52] A. Kaveh, S. Talatahari, and S. Talatahari, "Engineering optimization withhybrid particle swarm and ant colony optimization Set theoretical framework for meta-heuristic optimization algorithm View project ENGINEERING OPTIMIZATION WITH HYBRID PARTICLE SWARM AND ANT COLONY OPTIMIZATION," 2009. [Online]. Available: https://www.researchgate.net/publication/228667380

[53] C. A. Coello Coello and E. Mezura Montes, "Constraint-handling in genetic algorithms through the use of dominance-based tournament selection," *Advanced Engineering Informatics*, vol. 16, no. 3, pp. 193–203, Jul. 2002, doi: 10.1016/S1474-0346(02)00011-3.

Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023

[54] E. Sandgren, "Nonlinear Integer and Discrete Programming in Mechanical Design Optimization," *Journal of Mechanical Design*, vol. 112, no. 2, pp. 223–229, Jun. 1990, doi: 10.1115/1.2912596.

[55] C. A. Coello Coello, "Use of a self-adaptive penalty approach for engineering optimization problems," *Comput Ind*, vol. 41, no. 2, pp. 113–127, Mar. 2000, doi: 10.1016/S0166-3615(99)00046-9.

[56] S. M. Almufti, "Artificial Bee Colony Algorithm performances in solving Welded Beam Design problem," *Computer Integrated Manufacturing Systems*, vol. 28, no. 12, 2022, doi: 10.24297/j.cims.2022.12.17.

[57] A.-R. Hedar and M. Fukushima, "Derivative-Free Filter Simulated Annealing Method for Constrained Continuous Global Optimization," *Journal of Global Optimization*, vol. 35, no. 4, pp. 521–549, Aug. 2006, doi: 10.1007/s10898-005-3693-z.

[58] K. S. Lee and Z. W. Geem, "A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice," *Comput Methods Appl Mech Eng*, vol. 194, no. 36–38, pp. 3902–3933, Sep. 2005, doi: 10.1016/j.cma.2004.09.007.

[59] Algamal, Z. Y., Abonazel, M. R., & Lukman, A. F. (2023). Modified Jackknife Ridge Estimator for Beta Regression Model With Application to Chemical Data. International Journal of Mathematics, Statistics, and Computer Science, 1, 15–24. https://doi.org/10.59543/ijmscs.v1i.7713

[60] Obeid, N. (2023). On The Product and Ratio of Pareto and Erlang Random Variables. International Journal of Mathematics, Statistics, and Computer Science, 1, 33–47. https://doi.org/10.59543/ijmscs.v1i.7737

Doi: https://doi.org/10.54216/FPA.130102
Received: March 07, 2023 Revised: June 03, 2023 Accepted: August 09, 2023