Dynamic Leader Sibha Algorithm (DLSA): A Novel Hierarchical Metaheuristic Approach for Solving Engineering Design Problems

El-Sayed M. El-kenawy [1,2,3,*] , Amel Ali Alhussan[4], Doaa Sami Khafaga[4], Amal H. Alharbi[4], Sarah A. Alzakari[4], Abdelaziz A. Abdelhamid[5,6], Abdelhameed Ibrahim[1], Marwa M. Eid[7]

[1]School of ICT, Faculty of Engineering, Design and Information, Communications Technology (EDICT), Bahrain Polytechnic, PO Box 33349, Isa Town, Bahrain
[2]Applied Science Research Center. Applied Science Private University, Amman, Jordan
[3]Jadara University Research Center, Jadara University, Jordan
[4]Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia
[5]Department of Computer Science, Faculty of Computer and Information Sciences, Ain Shams University, Cairo 11566, Egypt
[6]Department of Computer Science, College of Computing and Information Technology, Shaqra University, 11961, Shaqra, Saudi Arabia
[7]Faculty of Artificial Intelligence, Delta University for Science and Technology, Mansoura 11152, Egypt

Emails: sayed.elkenawy@polytechnic.bh; aaalhussan@pnu.edu.sa; dskhafga@pnu.edu.sa; ahalharbi@pnu.edu.sa; saalzakari@pnu.edu.sa; abdelaziz@cis.asu.edu.eg; abdelhameed.fawzy@polytechnic.bh; mmm@ieee.org

Abstract

We present a new metaheuristic optimization technique, the Dynamic Leader Sibha Algorithm (DLSA), based on the structured dynamics of the ‘Sibha’ (an Islamic tool). Using a hierarchical leader-follower framework, DLSA dynamically balances exploration and exploitation to resolve the difficulties of high dimensional and multimodal optimization. DLSA is applied to three well-known engineering problems, namely the Speed Reducer, Welded Beam, and Pressure Vesseldo, to tackle the objectives of minimizing the weight of these structures and achieving the desired results with regularity. Key results indicate that DLSA is faster in convergence, gives better quality solutions and is more robust among diverse problem domains. DLSA is an effective and reliable optimization tool that can readily be applied to solve real-world and complex engineering problems.

Keywords: Dynamic Leader Sibha Algorithm; Metaheuristic Optimization; Engineering Design Problems; Exploration and Exploitation; High-Dimensional Search Spaces

1 Introduction

They abound in operations research, economics, finance, and artificial intelligence. Many problems contain high dimensionality, local optimum dominance, discontinuities and nonlinearity in the solution space. Linear programming, gradient-based techniques, dynamic programming, and many other traditional optimization techniques are often sufficient if the task is simple. Still, they severely underperform on complex tasks with multimodal, non-differentiable, or nonlinear optimal functions. Metaheuristic optimization methods have been developed and successfully applied as robust adaptive means for global optimization based on complex, possibly noise-corrupted and/or ill-defined problem domains as alternatives to the classical gradient-based approaches [1, 2].

DOI: https://doi.org/10.54216/JCIM.160110
Received: November 11, 2024 Revised: January 17, 2025 Accepted: February 19, 2025

The stochastic and heuristic principles of metaheuristic algorithms seem to aid in achieving a balance between exploration and exploitation, which are essential elements in any or most optimizations. Exploration prevents areas from being explored just once, allowing us to probe diverse regions of the solution space. At the same time, exploitation forces us to refine promising areas to converge to global optima. Achieving this optimal balance between these two goals is still a challenge. Over-exploration results in slow convergence, while excessive exploitation causes entrapment in local optima [3, 4]. However, the proposed metaheuristic algorithms suffer from scalability problems, premature convergence, and lack of consistent performance across various problem domains.

In the Dynamic Leader Sibha Algorithm (DLSA), we propose a new approach based on the structured counting mechanism of the ”Sibha”, an instrument used in Islamic religious activities. A hierarchical leader-follower framework is incorporated into the algorithm, characterized by solution agents moving dynamically based on fitness evaluation and adaptive movements. Existing algorithms are shown to have limitations that can be mitigated by structuring agent interactions and diversifying search patterns systematically. Unlike conventional swarm-based algorithms, which are highly reliant upon interpolation between neighboring individuals using randomness, DLSA uses a deterministic yet flexible leader-driven algorithm to achieve convergence reliability and robustness even in high-dimensional search spaces [5, 6].

DLSA's dynamic leadership mechanism is structured to approach the solution space. The leader agent explores high-potential regions, and follower agents dynamically update their trajectories according to the leader's location (relative position) and fitness. This hierarchical coordination minimizes redundant searches and efficiently explores complex multi-dimensional landscapes. Moreover, DLSA includes adaptive components that enable the search process to back away from low-quality regions to escape the detrimental traps of local optima and prevent the stagnation problem that occurs in many metaheuristics [7, 8].

In DLSA's structured leader-follower model, concerning the class of algorithms that employ stochastic randomization, global exploration and local exploitation are better balanced. Although these are randomized during initialization and introduce variability, the addition of Sibha-inspired dynamic adjustment offers agents a deterministic path, providing real-time evaluations of solution quality. This lowers our chance of premature convergence before reaching the optimal solution and increases the algorithm's capacity to solve highly dimensioned constraint problems [9, 10].

In this paper, DLSA is applied to solve three well-known engineering design optimization problems: the design problems of the Speed Reducer, Welded Beam, and Pressure Vessel. These problems form a rich testbed for evaluating the performance of DLSA as a diverse set of complex optimization problems with multimodal and high-dimensional constraints. As we address these problems, we hope to prove the versatility, efficiency, and, ultimately, the algorithm's effectiveness in obtaining high-quality solutions.

In this paper, we provide a complete description of the Dynamic Leader Sibha Algorithm and place it within the framework of all known metaheuristic optimization methods. We benchmark DLSA against well-known algorithms, including GA, PSO, and ACO, to measure its convergence speed, solution quality, and robustness in altering constraints. We highlight the advantages of DLSA and its potential to advance the adaptation of the next generation of adaptive, high-precision optimization algorithms [11–13], compared with other approaches.

This study finds that DLSA tackles the shortcomings of previously introduced algorithms and produces far better solutions than previously available algorithms, with superior operational reliability. The construction of DLSA provides a base for future works focusing on leader-driven structured methods for optimization to balance exploration and exploitation in complex solution spaces dynamically.

The main contributions of this study are as follows:

- **Novel Algorithm Development:** In this paper, we introduce the Dynamic Leader Sibha Algorithm (DLSA), based on Sibha's structured dynamics, which uses a similar hierarchical leader-follower mechanism to balance the exploration and exploitation aspects dynamically.
- **Engineering Applications:** DLSA is presented as the fully developed learning algorithm for solving three complex engineering design problems (Speed Reducer, Welded Beam, and Pressure Vessel) to illustrate its robustness and versatility.

DOI: https://doi.org/10.54216/JCIM.160110

Received: November 11, 2024 Revised: January 17, 2025 Accepted: February 19, 2025

- **Comparative Analysis:** We also conduct a detailed benchmarking of DLSA against state-of-the-art algorithms — GA, PSO, GWO, and WOA — proving its superiority in solution quality, convergence speed, and reliability.
- **Generalized Optimization Framework:** DLSA contains adaptable and dynamic mechanisms that constitute a scalable framework to solve many high-dimensional and non-linear optimization problems.
- **Future Potential:** The results enable the extension of DLSA to multi-objective optimization and hybridization with other methods, thus enabling the use of DLSA for various problems, including energy systems, supply chain optimization, and machine learning.

The structure of this paper is as follows. In Section 2, existing metaheuristic optimization algorithms are thoroughly reviewed regarding their strengths, limitations, and recent developments. Section 3 introduces the Dynamic Leader Sibha Algorithm (DLSA) with motivations, mathematical foundations, and a new hierarchical leader-follower mechanism. Additionally, this section details the application of DLSA to three benchmark engineering problems: Welded Beam, Pressure Vessel, and Speed Reducer. Finally, in Section 4, experimental results show that DLSA performs better than other optimization algorithms and that DLSA is adaptive, efficient, and converges in considerable cases. Section 5 concludes the study by summing up key contributions, exploring potential future applications, and suggesting future research directions.

2 Literature Review

Optimization is central to almost all disciplines with complex and high-dimensional challenges in engineering, data science, machine learning, and beyond. However, the familiar collapse of traditional methods in high-dimensional and multimodal cases has led to the rise of metaheuristic methods as adaptive and robust alternatives. Part II reviews recent advances in optimization via metaheuristic techniques, describing their motivation, results, and contributions towards addressing these challenges.

Multimodal Optimization (MO) techniques were investigated to face high-dimensional problems [14]. Inspired by the required methods to find multiple solutions within a single run, benchmark problems were adapted for Evolutionary Algorithm (EA)-based MO methods, including Dynamic Fitness Sharing (DFS) and local Best PSO. Results show that MO was able to find multiple high-quality solutions. The main contribution lies in setting MO as a general optimization landscape framework. Potential applications of the method involve the simultaneous optimization of conflicting objectives.

Motivated by the need to address discrete optimization challenges efficiently, the Binary Arithmetic Optimization Algorithm (BAOA) was introduced in [15]. The problem was mapped using a sigmoid transfer function to a "binary space," using the sigmoid to transform continuous solutions to the binary optimization problem. The results show better performance on 18 benchmark datasets compared with various algorithms (such as BDF and BPSO) regarding convergence speed and accuracy. It shows how BAOA's contribution lies in furnishing arithmetic operators for binary optimization in a novel manner, and it also demonstrates how this will help make such problems scalable in real-world applications.

Modified Multi-Swarm Particle Swarm Optimization (MSPSO) is introduced by [16] to increase swarm diversity and performance in high-dimensional space. The motivation was the limitations of standard PSO algorithms in terms of convergence and maintaining diversity. It was found that optimization of the SVM kernel parameters using MSPSO outperformed traditional PSO and GA. It presents an efficient multi-swarm framework for performing parameter tuning for machine learning tasks with competitive survival strategies.

Motivated by the fact that the existing binary optimizers are limited in classification tasks, the Whale Optimization Algorithm (WOA), as presented in [17], was adapted to binary spaces. Binary versions of WOA using Tournament and Roulette Wheel selection mechanisms improved the performance of benchmark problems. Classification accuracy and solution quality were superior to PSO, GA, and ALO. The study's main contribution is to extend the applicability of WOA to binary domains and optimization in data-driven settings. This was the motivation for Binary Brain Storm Optimization (BSO) as examined in [18]. Transfer functions were applied to transform real-valued solutions to binary, and then BSO was tested on complex optimization problems. BSO results showed competitive performance against state-of-the-art algorithms and reliable performance when several assumptions cannot be guaranteed. It is primarily robustly adaptable in binary domains, including for addressing discrete optimization problems.

Particle Swarm Optimization (PSO) was enhanced using a bare-bones framework and unique local search operators to overcome premature convergence [19]. The algorithm augmented exploitation with additional and delete operators to foster convergence in higher-dimensional spaces. Their results validated its superior performance compared to eleven state-of-the-art algorithms. Specifically, its contribution is the refinement of PSO for precision applications, which has implications for complex engineering problems that demand high accuracy.

In [20], Unsupervised Optimization based on Ant Colony Optimization (UACO) was proposed for unsupervised optimization. Given the limitations of supervised methods in feature selection, we (via UACO) used feature similarity without given labels. Results showed its efficiency and low computational complexity, and it was found to be more efficient than other methods when working on high-dimensional tasks. The study's contribution is to provide an innovative, unsupervised framework that can be applied in data mining or any machine learning application.

In [21], dynamic optimization was explored with Moth-Flame Optimization (MFO) based on moth motion patterns. This was motivated by the necessity for accurate and reliable optimization techniques. Results showed that MFO performs far better than the standard algorithms such as PSO and GA. In particular, it contributes a wrapper-based approach to solving high-dimensional optimization problems and discusses the implications for computationally intensive problems.

Diverse optimization challenges inspired the Mountain Gazelle Optimizer (MGO) introduced in [22], based on gazelle social structures. For the sake of scalability of the algorithms, MGO outperformed 9 other tested algorithms on 52 benchmark functions. Its contribution lies in harnessing natural hierarchies for optimization, which holds promise for engineering and data-driven applications demanding flexibility and precision.

A comprehensive review of the Multi-Verse Optimizer (MVO) [23] was provided since it is versatile in the different optimization domains. Variations like binary, chaotic, and multi-objective MVO were analyzed, and their flexibility and effectiveness were exposed. Results demonstrate MVO's applicability to problems of interest beyond network optimization and parameter tuning. Its primary contribution is its adaptability to complex optimization tasks, drawing implications for advancing metaheuristic research.

This review focuses on various metaheuristic optimization advancements featuring algorithms that explore and exploit to overcome the classical challenges. The review exhibits how these techniques address the problems of high dimensions and multimodality within the MRCI framework, thereby serving as a prerequisite to the continued evolution of this field.

3 Proposed Dynamic Leader Sibha Algorithm (DLSA)

3.1 Inspiration

The inspiration behind the Dynamic Leader Sibha Algorithm (DLSA) comes from the structured counting mechanism of the "Sibha," traditionally used in Islamic practices for counting. Embracing this notion, a hierarchical leader-follower framework is translated, which emulates structured organization and systematic movement within the search space.

In this way, the dynamic leadership mechanism inspired by the Sibha ensures that search agents work under the control of a dynamically chosen leader, which enhances collaboration and efficiency in tackling complex optimization landscapes. DLSA structures the interaction between agents to prevent random wandering, steering the search toward fruitful regions while sustaining diversity.

This hierarchical leader-follower framework addresses two critical challenges in optimization: stagnation and premature convergence. By adopting the dynamic leadership strategy, the algorithm can react to the problem

DOI: https://doi.org/10.54216/JCIM.160110

Received: November 11, 2024 Revised: January 17, 2025 Accepted: February 19, 2025

landscape, pivoting from low-quality regions while refining high-potential regions. The structured interaction between agents accommodates a systematic solution space exploration, eliminating redundancy and increasing convergence reliability.

With this new approach, DLSA balances exploration and exploitation and provides a robust, adaptive way to handle high-dimensional, multimodal optimization problems.

3.2 Mathematical Foundation of DLSA

3.2.1 Exploration

The exploration phase in the Dynamic Leader Sibha Algorithm (DLSA) focuses on global search, ensuring that diverse regions of the solution space are investigated. The position of the current search agent, $\vec{W}(t+1)$, is updated using the following equation:

$$\vec{W}(t+1) = \vec{W}(t) + K\left(\vec{L}(t) - \vec{W}(t)\right)$$

Here, $\vec{L}(t)$ represents the leader (best search agent) at iteration t, and K is a parameter controlling the step size based on the difference between the leader and the current agent. This facilitates guided exploration toward promising regions of the solution space.

The leader's position, $\vec{L}(s+1)$, is further refined as:

$$\vec{L}(s+1) = \vec{L}(s) + Z\left(\vec{W}(t+1) - \vec{P}_D\right) + g\left(\frac{\vec{L}(s) + \vec{W}(t+1)}{K}\right)$$

Where $\vec{P}_D$ represents the previous search agent. The parameter Z exponentially increases from 1 to 2, dynamically controlling the influence of the difference between the current position and $\vec{P}_D$. The term g adapts the search direction, calculated as:

$$g = \left(\frac{k \times 2\sin(\theta)}{Z} + \frac{k+Z}{K}\right) \times iteration$$

The summation-based calculation of K ensures adaptive control over the search dynamics:

$$K = \sum_{n=0}^{n=8}\left(\frac{\sin\Theta}{Z}\right), \quad \Theta \in (0, 4\pi)$$

These equations enable a controlled and systematic search space exploration, minimizing the risk of stagnation and enhancing convergence reliability.

3.2.2 Exploitation

The exploitation phase refines the search around promising areas, focusing on local optimization. The leader's position, $L(s+1)$, is updated using:

$$L(s+1) = \frac{1}{K}\sum_{n=0}^{Z}\left(\frac{\vec{W}(t+1) + \vec{L}(s)}{g}\right)^n \times \left(\vec{L}(s) - \vec{P}_p\right)$$

Here, $\vec{P}_p$ represents the position of a peer agent, and the summation ensures that the search prioritizes regions with high potential while retaining diversity. This phase narrows the focus to regions with promising fitness values, improving the precision of the solution.

DOI: https://doi.org/10.54216/JCIM.160110

Received: November 11, 2024 Revised: January 17, 2025 Accepted: February 19, 2025

3.2.3 Elimination (Elas)

The elimination phase dynamically adjusts agents' positions to refine the solution space further. The leader's position is updated as follows:

$$L(s+1) = K\left(\frac{\vec{L}(s)+\vec{W}(t)}{Z\times g\times K}\right)+g\left(\frac{\vec{L}(s)+\vec{P}_D}{Z\times g\times K}\right)+Z\left(\frac{\vec{W}(t)+\vec{P}_D}{Z\times g\times K}\right)$$

In this phase, we utilize hierarchical leader-follower dynamics to prune suboptimal solutions, guiding further exploration in proximity to promising regions.

3.2.4 Additional Notations

- $\vec{L}(s+1)$: New leader position at the $s+1$-th iteration.
- $\vec{W}(t+1)$: New position of the current search agent.
- g: The method for adjusting parameters with ensuing iteration and exploration factors.
- K: Control parameter for search dynamics defined in terms of summation.
- Z: Growth parameter in the search direction.

These components allow DLSA to achieve a good exploration-exploitation tradeoff, robustly optimizing various problems.

3.3 Engineering Problems and Their Equations

To evaluate the performance of the Dynamic Leader Sibha Algorithm (DLSA), three complex engineering design problems are considered: the Speed Reducer, Welded Beam, and Pressure Vessel problems. Due to their nonlinear constraints, multidimensional search spaces, and conflicting objectives, these problems constitute benchmarks in optimization research. The problems are distinct and challenge the algorithm to solve them optimally, subject to tight constraints. The details of each problem are provided below in terms of the objective function, constraints, and variable bounds.

3.3.1 Speed Reducer Problem

In the Speed Reducer problem, we study a mechanical speed reducer whose cost is to be minimized, subject to bending stress, contact stress, and geometric dimension constraints. The objective function is given as:

$$f(\vec{x}) = 0.7854x_1x_2^2\left(3.3333x_3^2+14.9334x_3-43.0934\right)-1.508x_1(x_6^2+x_7^2)+7.4777(x_6^3+x_7^3)+0.7854(x_4x_6^2+x_5x_7^2)$$

Subject to the following constraints:

Received: November 11, 2024 Revised: January 17, 2025 Accepted: February 19, 2025

$$
\begin{aligned}
g_1(\vec{x}) &= \frac{27}{x_1x_2^2x_3} - 1 \leq 0, \quad g_2(\vec{x}) = \frac{397.5}{x_1x_2^2x_3^2} - 1 \leq 0, \\
g_3(\vec{x}) &= \frac{1.93x_4^3}{x_2x_3x_6^4} - 1 \leq 0, \quad g_4(\vec{x}) = \frac{1.93x_5^3}{x_2x_3x_7^4} - 1 \leq 0, \\
g_5(\vec{x}) &= \frac{1}{110x_6^3}\sqrt{\left(\frac{745.0x_4}{x_2x_3}\right)^2 + 16.9 \times 10^6} - 1 \leq 0, \\
g_6(\vec{x}) &= \frac{1}{85x_7^3}\sqrt{\left(\frac{745.0x_5}{x_2x_3}\right)^2 + 157.5 \times 10^6} - 1 \leq 0, \\
g_7(\vec{x}) &= \frac{x_2x_3}{40} - 1 \leq 0, \quad g_8(\vec{x}) = \frac{5x_2}{x_1} - 1 \leq 0, \\
g_9(\vec{x}) &= \frac{x_1}{12x_2} - 1 \leq 0, \quad g_{10}(\vec{x}) = \frac{1.5x_6 + 1.9}{x_4} - 1 \leq 0, \\
g_{11}(\vec{x}) &= \frac{1.1x_7 + 1.9}{x_5} - 1 \leq 0.
\end{aligned}
$$

Variable bounds are defined as follows:

$$
\begin{aligned}
&2.6 \leq x_1 \leq 3.6, \quad 0.7 \leq x_2 \leq 0.8, \quad 17 \leq x_3 \leq 28, \\
&7.3 \leq x_4, x_5 \leq 8.3, \quad 2.9 \leq x_6 \leq 3.9, \quad 5.0 \leq x_7 \leq 5.5.
\end{aligned}
$$

The Speed Reducer Design Problem is concerned with optimizing the design of a mechanical speed reducer, given constraints on stress, deflection, and geometric parameters, to minimize the manufacturing cost. Figure 1 shows a schematic of the speed reducer with essential features like the gear tooth model (X_2), the number of teeth (X_3), and the shaft and bearing dimensions (X_1, X_4, X_5, X_6, X_7). It illustrates the trade-off between cost minimization and mechanical performance, providing practical engineering solutions.

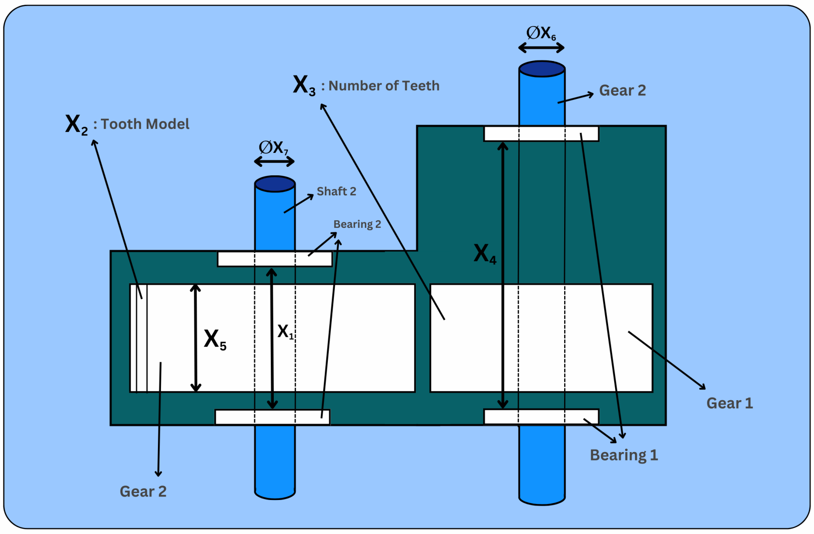

Figure 1: Schematic of the Speed Reducer Design Problem.

DOI: https://doi.org/10.54216/JCIM.160110

Received: November 11, 2024 Revised: January 17, 2025 Accepted: February 19, 2025

3.3.2 Welded Beam Problem

The objective is to minimize the fabrication cost of a beam subject to constraints on shear stress, bending stress, deflection, and weld size. The objective function is defined as:

$$f(w, L, d, h) = 1.10471w^2L + 0.04811dh(14.0 + L)$$

Subject to the following constraints:

$$\begin{aligned}
&g_1 = w - h \leq 0, \quad g_2 = \delta - 0.25 \leq 0, \quad g_3 = \tau - 13,600 \leq 0, \\
&g_4 = \sigma - 30,000 \leq 0, \quad g_5 = 0.125 - w \leq 0, \quad g_6 = 6000 - P \leq 0, \\
&g_7 = 0.10471w^2 + 0.04811hd(14 + L) - 0.5 \leq 0.
\end{aligned}$$

Where:

$$\sigma = \frac{504,000}{hd^2}, \quad Q = 6000\left(14 + \frac{L}{2}\right), \quad D = \frac{1}{2}\sqrt{L^2 + (w+d)^2},$$

$$J = \sqrt{2}wL\left(\frac{L^2}{6} + \frac{(w+d)^2}{2}\right), \quad \delta = \frac{65856}{30000hD^3}, \quad \tau = \sqrt{\alpha^2 + \left(\frac{\alpha\beta L}{D}\right)^2 + \beta^2},$$

$$\alpha = \frac{6000}{\sqrt{2}wL}, \quad \beta = \frac{QD}{J}, \quad P = 0.61432 \times 10^6 \frac{dh^3}{6}\left(1 - \frac{d\sqrt{30/48}}{28}\right).$$

Variable bounds are defined as:

$$0.1 \leq w, h \leq 2.0, \quad 0.1 \leq L, d \leq 10.$$

The Welded Beam Design Problem consists of optimizing the design of a welded beam structure to minimize the cost of fabrication, subject to constraints on shear stress, bending stress, and deflection of the structure. Figure 2 shows the key design parameters, such as the weld thickness (w), length of the beam (L), depth of the beam (d), and height of the beam (h). The external load (P) applied to the beam is also shown in the figure. The practical importance lies in balancing structural integrity with cost efficiency.

DOI: https://doi.org/10.54216/JCIM.160110

Received: November 11, 2024 Revised: January 17, 2025 Accepted: February 19, 2025

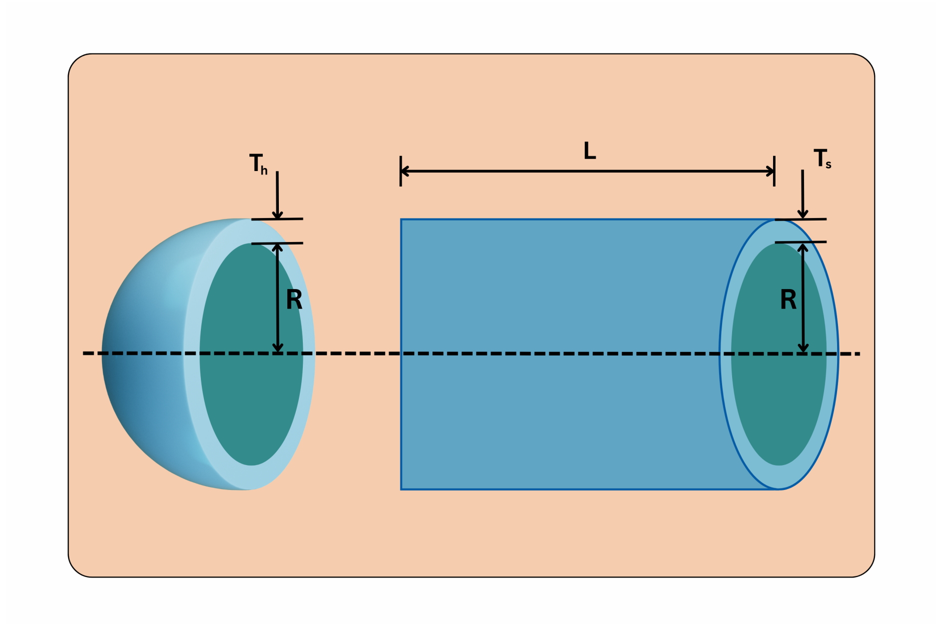

Figure 2: Schematic of the Welded Beam Design Problem.

3.3.3 Pressure Vessel Problem

The Pressure Vessel problem is to minimize the cost of designing a cylindrical pressure vessel, subject to constraints on stress, thickness, and volume. The objective function is:

$$f(T_s, T_h, R, L) = 0.6224T_sRL + 1.7781T_hR^2 + 3.1661T_s^2L + 19.84T_h^2L.$$

Subject to the constraints:

$$g_1 = -T_s + 0.0193R \leq 0, \quad g_2 = -T_h + 0.0095R \leq 0,$$
$$g_3 = -\pi R^2L - \frac{4}{3}\pi R^3 + 1,296,000 \leq 0, \quad g_4 = L - 240 \leq 0.$$

Variable bounds are defined as:

$$0 \leq T_s, T_h \leq 99, \quad 10 \leq R, L \leq 200.$$

Evaluation Criteria:

For each problem, the evaluation focuses on:

- Satisfying all nonlinear constraints.
- The performance metrics of optimal cost, convergence speed, and computational efficiency are compared across different optimization algorithms.

DOI: https://doi.org/10.54216/JCIM.160110

Received: November 11, 2024 Revised: January 17, 2025 Accepted: February 19, 2025

The optimization problem of the cost of a cylindrical pressure vessel, subject to constraints on material strength, thickness, and volume, is the Pressure Vessel Design Problem. Figure 3 shows the geometry of the pressure vessel with the shell thickness (T_s), head thickness (T_h), radius (R), and length of the cylindrical section (L). Ensuring safety and functionality at minimum manufacturing cost is critical to structural and industrial design and is well emphasized by this problem.

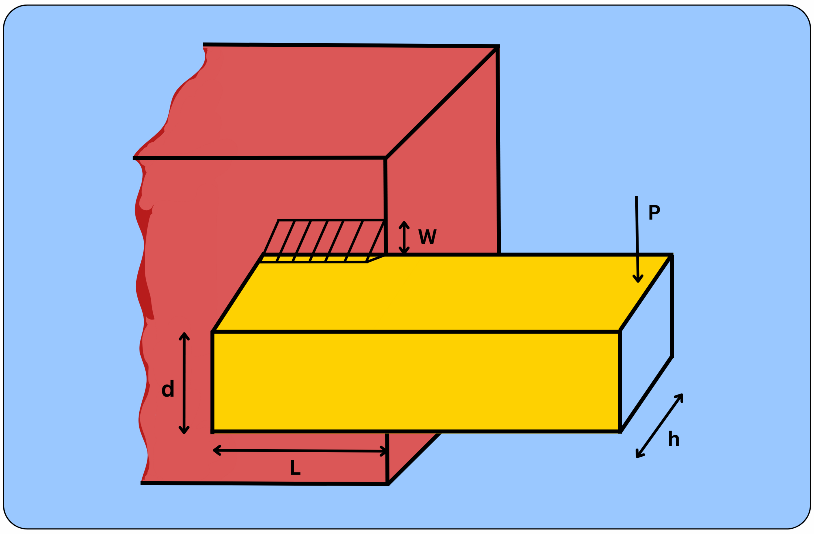

Figure 3: Schematic of the Pressure Vessel Design Problem.

Including these problems highlights DLSA's generality in solving nontrivial, real-world optimization tasks.

4 Results and Discussion

4.1 Parameters of the Algorithm

The DLSA is based on three key parameters that dynamically control the search process by balancing exploration and exploitation. These parameters are detailed in Table 1.

Table 1: Key Parameters of the Dynamic Leader Sibha Algorithm (DLSA)

Parameter	Description
K	Calculated as $K = \sum_{n=0}^{n=8} \left(\frac{\sin\Theta}{Z}\right)$, controlling the step size and adaptability in the exploration phase.
Θ	A dynamic angle ranging between 0 and 4π, used in the computation of K.
Z	A parameter that increases exponentially from 1 to 2, governing the adaptability and influencing the search direction.

These settings were consistent across all test problems, ensuring fair comparisons with other algorithms.

DOI: https://doi.org/10.54216/JCIM.160110

Received: November 11, 2024 Revised: January 17, 2025 Accepted: February 19, 2025

4.2 Performance on Engineering Problems

To evaluate the effectiveness of DLSA, three engineering design problems were considered: the Speed Reducer, Welded Beam, and Pressure Vessel problems. These benchmarks have widely been used in optimization research because they feature nonlinear constraints, conflicting objectives, and high-dimensional search spaces. A new series of benchmark problems for numerical optimization is described.

1. Speed Reducer Problem:

The Speed Reducer problem seeks to reduce the cost of a mechanical speed reducer, subject to constraints on stress, deflection, and geometry. The multimodal objective function and strict feasibility requirements make this a critical challenge in the engineering design problem. Table 2 presents the optimal solutions obtained by DLSA and other algorithms. DLSA attained the lowest optimal cost, outperforming other approaches in clearing ambiguous and complex solution spaces.

Table 2: Speed Reducer Engineering Problem Results

Algorithm	X_1	X_2	X_3	X_4	X_5	X_6	X_7	Optimal Cost
DLSA	3.5019	0.7	17	7.3392	7.8829	3.3523	5.2868	2999.8243
GA	3.5046	0.7	17	7.3	7.8	3.3568	5.2878	3000.5645
GWO	3.5028	0.7	17.0081	7.3435	7.8076	3.3643	5.2879	3003.7862
WOA	3.5000	0.7	17	7.3	7.8	3.4260	5.2940	3020.7574
PSO	3.6	0.7	17	7.3	7.8	3.3502	5.2867	3035.6256

Table 3 presents the statistical results for the Speed Reducer problem, including the average fitness, standard deviation, and best solution obtained across multiple runs. DLSA achieved the best solution and exhibited high consistency with the lowest standard deviation, highlighting its robustness and reliability.

Table 3: Statistical Results of the Speed Reducer Engineering Problem

Algorithm	Average Fitness	Standard Deviation	Best Solution
DLSA	3004.4238	3.5744	2999.0703
GA	97064826240.6252	520531224338.84	2998.1889
GWO	3014.0967	5.3480	3003.9460
PSO	3110.3109	83.9934	3035.6256
WOA	3654.3691	719.3737	3048.5743

2. Welded Beam Problem:

The Welded Beam problem involves minimizing the fabrication cost of a beam while adhering to constraints on shear stress, bending stress, and deflection. This problem is particularly challenging due to its intricate constraint handling and large feasible region. Table 4 illustrates the results, where DLSA outperformed other algorithms in achieving the lowest optimal cost.

Table 4: Welded Beam Engineering Problem Results

Algorithm	X_1	X_2	X_3	X_4	Optimal Cost
DLSA	0.2057	3.4705	9.0366	0.2057	1.7249
GWO	0.2049	3.4926	9.0349	0.2060	1.7280
PSO	0.2005	3.5792	9.0790	0.2056	1.7373
GA	0.2115	3.4548	8.8988	0.2122	1.7568
WOA	0.2093	3.4146	8.9863	0.2154	1.7866

The statistical results for the Welded Beam problem are presented in Table 5. DLSA demonstrated superior consistency with the lowest standard deviation and high reliability in finding the best solution.

DOI: https://doi.org/10.54216/JCIM.160110
Received: November 11, 2024 Revised: January 17, 2025 Accepted: February 19, 2025

Table 5: Statistical Results of the Welded Beam Engineering Problem

Algorithm	Average Fitness	Standard Deviation	Best Solution
DLSA	1.7896	0.0593	1.7264
GA	2.4362	0.5731	1.7411
GWO	1.7303	0.0020	1.7266
PSO	1.8837	0.8074	1.7249
WOA	2.7330	0.6605	1.8550

3. Pressure Vessel Problem:

The Pressure Vessel problem seeks to minimize the cost of designing a cylindrical pressure vessel while satisfying constraints on stress and volume. This problem is a standard benchmark in structural optimization. Table 6 summarizes the results, where DLSA outperformed other algorithms by achieving the lowest cost.

Table 6: Pressure Vessel Engineering Problem Results

Algorithm	T_s	T_h	R	L	Optimal Cost
DLSA	0.7782	0.3847	40.3197	200	5885.4664
DE	0.7791	0.3850	40.3333	199.8787	5892.7035
PSO	0.8002	0.3955	41.4606	184.7042	5924.0488
GWO	0.7812	0.3940	40.3991	200	5947.1459
GA	0.8469	0.4204	43.6351	159.7470	6081.0415
WOA	1.0608	0.5298	51.8960	83.9799	6872.2534

Table 7 presents the statistical results for the Pressure Vessel problem. DLSA exhibited high robustness with the best average fitness and the smallest standard deviation.

Table 7: Statistical Results of the Pressure Vessel Engineering Problem

Algorithm	Best	Average	Standard Deviation
DLSA	5889.1852	6029.3344	237.3213
GA	5959.1476	6857.1012	530.4589
GWO	5963.1878	6170.1240	290.0529
DE	5885.6793	6010.1931	355.6939
PSO	5957.6601	6155.2294	154.1526
WOA	6648.0565	13705.2869	12906.0826

The results confirm that the Dynamic Leader Sibha Algorithm (DLSA) demonstrates significant advantages over other algorithms across the three benchmark engineering problems: The three parts listed are Speed Reducer, Welded Beam, and Pressure Vessel. For all three problems, DLSA consistently found the best solution solution. In addition, the algorithm was highly consistent and robust across several runs with low standard deviation for statistical results. Its versatility as a robust optimization framework capable of solving diverse engineering design problems is due to its adaptability and efficiency in handling constraints and nonlinear objectives.

Results from this study validate the DLSA optimization algorithm as state-of-the-art. DLSA has demonstrated superior performance in optimality of the achieved solution, convergence behavior, and robustness over a wide range of challenging engineering problems, thus making itself a reliable and versatile tool for real-world applications. It performs exploration and exploitation using its hierarchical leader-follower mechanism and dynamic adaptability, contributing to metaheuristic optimization. Further, future research can extend its applicability to multi-objective and hybrid optimization scenarios.

DOI: https://doi.org/10.54216/JCIM.160110

Received: November 11, 2024 Revised: January 17, 2025 Accepted: February 19, 2025

5 Conclusion

This paper introduced a novel metaheuristic optimization algorithm inspired by the structured dynamics of Sibha, known as the dynamic leader Sibha Algorithm (DLSA). This paper shows that using a hierarchical leader-follower framework, DLSA balances exploration and exploitation that tackles key problems in high dimensional, nonlinear optimization problems. It affords robust and convergent search through complex search landscapes, which it does at a price of dynamic adaptability so that progress is never stalled.

The algorithm's performance was validated on three benchmark engineering problems: Speed Reducer, Welded Beam, and Pressure Vessel. DLSA outperformed, and consistently outperformed, all state-of-the-art algorithms, providing the best set of solutions with high reliability and low variability across multiple runs. The results demonstrate that it is a robust and efficient optimization tool for real-world applications.

The research can continue extending DLSA to handle multi-objective cases and hybridizing it with other algorithms to increase scalability. Also, its adaptability makes it a promising candidate for applications in energy systems, supply chain management, and machine learning. DLSA makes a solid contribution to the field of metaheuristic optimization and paves the way for subsequent advances in metaheuristic optimization.

References

[1] K. S. Guedes, C. F. de Andrade, P. A. C. Rocha, R. dos S. Mangueira, and E. P. de Moura. Performance analysis of metaheuristic optimization algorithms in estimating the parameters of several wind speed distributions. *Applied Energy*, 268:114952, 2020.

[2] A. Mohammadzadeh, M. Masdari, F. S. Gharehchopogh, and A. Jafarian. A hybrid multi-objective metaheuristic optimization algorithm for scientific workflow scheduling. *Cluster Computing*, 24(2):1479–1503, 2021.

[3] M. Dehghani and H. Samet. Momentum search algorithm: A new meta-heuristic optimization algorithm inspired by momentum conservation law. *SN Applied Sciences*, 2(10):1720, 2020.

[4] M. Cikan and B. Kekezoglu. Comparison of metaheuristic optimization techniques including equilibrium optimizer algorithm in power distribution network reconfiguration. *Alexandria Engineering Journal*, 61(2):991–1031, 2022.

[5] J. Zhou, X. Shen, Y. Qiu, X. Shi, and M. Khandelwal. Cross-correlation stacking-based microseismic source location using three metaheuristic optimization algorithms. *Tunnelling and Underground Space Technology*, 126:104570, 2022.

[6] F. N. Al-Wesabi, M. Obayya, M. A. Hamza, J. S. Alzahrani, D. Gupta, and S. Kumar. Energy aware resource optimization using unified metaheuristic optimization algorithm allocation for cloud computing environment. *Sustainable Computing: Informatics and Systems*, 35:100686, 2022.

[7] F. A. Hashim, K. Hussain, E. H. Houssein, M. S. Mabrouk, and W. Al-Atabany. Archimedes optimization algorithm: A new metaheuristic algorithm for solving optimization problems. *Applied Intelligence*, 51(3):1531–1551, 2021.

[8] M. Hamza Zafar, N. Mujeeb Khan, A. Feroz Mirza, M. Mansoor, N. Akhtar, M. Usman Qadir, N. Ali Khan, and S. K. Raza Moosavi. A novel meta-heuristic optimization algorithm based mppt control technique for pv systems under complex partial shading condition. *Sustainable Energy Technologies and Assessments*, 47:101367, 2021.

[9] M. Dehghani and H. Samet. Momentum search algorithm: A new meta-heuristic optimization algorithm inspired by momentum conservation law. *SN Applied Sciences*, 2(10):1720, 2020.

[10] A. H. Halim, I. Ismail, and S. Das. Performance assessment of the metaheuristic optimization algorithms: An exhaustive review. *Artificial Intelligence Review*, 54(3):2323–2409, 2021.

DOI: https://doi.org/10.54216/JCIM.160110

Received: November 11, 2024 Revised: January 17, 2025 Accepted: February 19, 2025

[11] M. H. Qais, H. M. Hasanien, and S. Alghuwainem. Transient search optimization: A new meta-heuristic optimization algorithm. *Applied Intelligence*, 50(11):3926–3941, 2020.

[12] H. Tran-Ngoc, S. Khatir, H. Ho-Khac, G. De Roeck, T. Bui-Tien, and M. Abdel Wahab. Efficient artificial neural networks based on a hybrid metaheuristic optimization algorithm for damage detection in laminated composite structures. *Composite Structures*, 262:113339, 2021.

[13] M. A. Hamza J. S. Alzahrani D. Gupta F. N. Al-Wesabi, M. Obayya and S. Kumar. Energy aware resource optimization using unified metaheuristic optimization algorithm allocation for cloud computing environment. *Sustainable Computing: Informatics and Systems*, 35:100686, 2022.

[14] S. Kamyab and M. Eftekhari. Feature selection using multimodal optimization techniques. *Neurocomputing*, 171:586–597, 2016.

[15] N. Khodadadi, E. Khodadadi, Q. Al-Tashi, E.-S. M. El-Kenawy, L. Abualigah, S. J. Abdulkadir, A. Alqushaibi, and S. Mirjalili. Baoa: Binary arithmetic optimization algorithm with k-nearest neighbor classifier for feature selection. *IEEE Access*, 11:94094–94115, 2023.

[16] Y. Liu, G. Wang, H. Chen, H. Dong, X. Zhu, and S. Wang. An improved particle swarm optimization for feature selection. *Journal of Bionic Engineering*, 8(2):191–200, 2011.

[17] M. Mafarja and S. Mirjalili. Whale optimization approaches for wrapper feature selection. *Applied Soft Computing*, 62:441–453, 2018.

[18] J. P. Papa, G. H. Rosa, A. N. de Souza, and L. C. S. Afonso. Feature selection through binary brain storm optimization. *Computers & Electrical Engineering*, 72:468–481, 2018.

[19] X. Song, Y. Zhang, D. Gong, and X. Sun. Feature selection using bare-bones particle swarm optimization with mutual information. *Pattern Recognition*, 112:107804, 2021.

[20] S. Tabakhi, P. Moradi, and F. Akhlaghian. An unsupervised feature selection algorithm based on ant colony optimization. *Engineering Applications of Artificial Intelligence*, 32:112–123, 2014.

[21] H. M. Zawbaa, E. Emary, B. Parv, and M. Sharawi. Feature selection approach based on moth-flame optimization algorithm. In *2016 IEEE Congress on Evolutionary Computation (CEC)*, pages 4612–4617, 2016.

[22] B. Abdollahzadeh, F. S. Gharehchopogh, N. Khodadadi, and S. Mirjalili. Mountain gazelle optimizer: A new nature-inspired metaheuristic algorithm for global optimization problems. *Advances in Engineering Software*, 174:103282, 2022.

[23] V. P. Singh A. K. Sangaiah and M. Shojafar. Metaheuristic algorithms in industry 4.0: Recent advances, issues, and applications. *Computers Electrical Engineering*, 89:106903, 2020.

DOI: https://doi.org/10.54216/JCIM.160110

Received: November 11, 2024 Revised: January 17, 2025 Accepted: February 19, 2025