QUBAHAN TECHNO JOURNAL
VOL. 2, NO. 2, May 2023
https://doi.org/10.48161/qtj.v2n2a19

Comprehensive Review and Hybrid Evolution of Teaching–Learning-Based Optimization

Nodira R. Rustamova [1], Raveenthiran Vivekanantharasa [2]

[1]Department of Psychology and Pedagogy, International School of Finance Technology and Science (Private University), Tashkent 100047, Uzbekistan
[2]Faculty of Education, The Open University of Sri Lanka, Sri Lanka

ABSTRACT: Teaching–Learning-Based Optimization (TLBO) stands out as a novel population-based metaheuristic inspired by the pedagogical process in a classroom, in which a teacher imparts knowledge to learners and the learners enhance their performance by mutual interaction. This paper provides a comprehensive review of TLBO along with its extensive hybrid and intelligent extensions developed over recent years. We examine the fundamental algorithmic principles of TLBO—its parameter-free nature, two-phase (teacher and learner) approach, and inherent simplicity—and contrast its performance across a range of benchmarks and real-world engineering optimization problems. In addition, we survey various taxonomic categories such as adaptive TLBO, multi-objective TLBO, discrete variants, and hybrids that integrate TLBO with other metaheuristics including Genetic Algorithms (GA), Particle Swarm Optimization (PSO), Robust Tabu Search (RTS), and Harmony Search (HS). Special attention is given to recent innovations in hybrid frameworks, such as the TLBO–RTS and TLBO–CO algorithms, where complementary search techniques enhance global exploration while preserving the rapid convergence property. Finally, the paper discusses theoretical aspects including convergence properties, computational complexity, and outlines current challenges and promising directions for future research.

KEYWORDS: Teaching–Learning-Based Optimization, Metaheuristics, Optimization, Hybrid algorithm.

I. INTRODUCTION

Metaheuristic optimization algorithms have become critical tools for solving complex, multi-dimensional problems across engineering, energy systems, machine learning, and operations research. Among numerous methods, Teaching–Learning-Based Optimization (TLBO) has garnered significant interest due to its simplicity, lack of algorithm-specific parameters, and potent capability to converge rapidly toward high-quality solutions6. Inspired by the classroom teaching–learning process, TLBO models the optimization process via two distinct phases: the teacher phase, where the best solution guides the population toward improved performance, and the learner phase, in which individuals further refine their solutions through mutual learning and interaction[1].

TLBO's parameter-free nature is particularly appealing when compared with other swarm intelligence algorithms, which usually require time-consuming tuning of several parameters (e.g., inertia weight, crossover and mutation rates)[2]. Over time, numerous research studies have extended the canonical TLBO into various derivative forms addressing multi-objectivity, discrete variables, and the challenges posed by high-dimensional search spaces. Hybrid frameworks that combine TLBO with other metaheuristics—such as Genetic Algorithms (GA), Particle Swarm Optimization (PSO), and advanced local search engines like Robust Tabu Search (RTS)—have further enhanced its applicability in diverse domains[3], [4], [5].

This article aims to present a comprehensive review and hybrid evolution of Teaching–Learning-Based Optimization. It covers the fundamental algorithmic structure, reviews numerous variants developed in the literature, presents performance comparisons on benchmark problems, and discusses applications in engineering optimization. The article further examines recent state-of-the-art hybrid frameworks that incorporate TLBO into complex multi-objective and high-dimensional problem solving. By critically analyzing both the strengths and limitations of TLBO and its extensions, we aim to provide a valuable resource for researchers and practitioners engaged in the field of metaheuristic optimization[6].

II. OVERVIEW OF METAHEURISTICS

Metaheuristics are high-level problem-independent algorithmic frameworks that provide approximate solutions for computationally complex optimization problems. They have been successfully applied to both continuous and discrete problem spaces. The key strengths of metaheuristics include[7]:

- Exploration versus Exploitation: Metaheuristics typically balance exploring new regions of the solution space (diversification) and intensively searching locally for the best solution (intensification or exploitation). This dual process is central to their performance despite the absence of gradient information.
- Population-Based Approaches: Many metaheuristics, including Particle Swarm Optimization (PSO), Genetic Algorithms (GA), and the Firefly Algorithm (FA), use a population of candidate solutions that evolve over time toward better solutions. This shared learning model often leads to robust convergence properties. See figure 1.
- Adaptability to Complex Landscapes: These algorithms have been employed in multidimensional and noisy environments where analytical gradient-based methods might fail, making them particularly valuable in real-world applications such as scheduling, design optimization, and machine learning.

Metaheuristic methods are continuously adapted and hybridized to overcome their inherent limitations, and the Firefly Algorithm is a prominent example of such innovation in swarm intelligence.

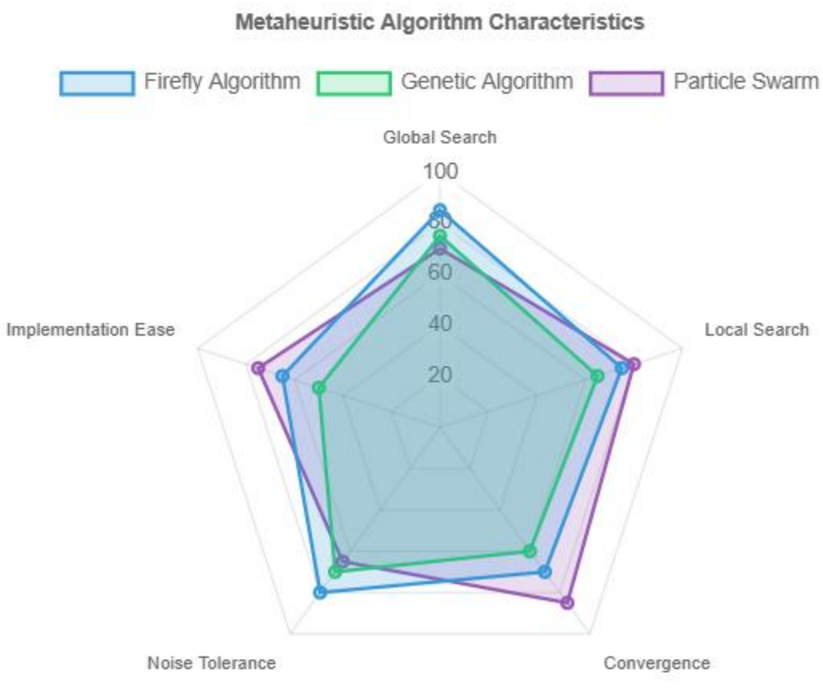

Figure 1: population based algorithm characteristics

According to Almufti in 2019[8], there were more than 200 Metaheuristic algorithms have been developed to address a wide range of practical problems. Most algorithms get their inspiration from nature and incorporate elements of physical, biological, ethological, or swarm intelligence. Surprisingly, several of these methods, such as the Vibrating Particles System (VPS), particle swarm optimization (PSO), Ant Colony Algorithm(ACO), Social Spider Optimization (SSO), Water Evaporation Optimization (WEO), and Big Bang-Big Crunch Algorithm (BB-BC) are well recognized among specialists from other study domains in addition to computer scientists. In actuality, the widespread use of metaheuristic algorithms, particularly in engineering optimization problems, may be attributed to a number of factors, including their flexibility, gradient-free mechanism, and reliance on basic ideas for local optima avoidance. Because they are frequently based on plain or simple notions, the majority of nature-inspired metaheuristics algorithms are almost simple[9].

Metaheuristic algorithms are broadly grouped into different categories: evolutionary, swarm, physics or chemistry, and human behavior based algorithms. Evolutionary algorithms are just an adaptation to natural evolutionary processes. The global optima are obtained in this type of algorithm by producing a new child that inherits the features and properties of parents by randomly selecting agents from the present population as parents and involving them in the production of offspring for the next generation. Common evolutionary-based algorithms include evolutionary strategies, genetic algorithms , and genetic programming (GP), ant colony optimization (ACO). Though they addressed various optimization issues, such as the infinite monkey theorem, Richard Dawkins' weasel, and the travelling salesman problem, the fundamental disadvantage of these methods is their computational cost.

Swarm based algorithms replicate the social and intellectual behaviour of a bunch of species (e.g., birds, insects, fish). For example, the famous particle swarm optimization technique was inspired by bird flight, while a new moth-flame optimization approach was inspired by moth navigation. Swarm-based algorithms tackle optimization problems by exhibiting self-organization, resilience, coordination, simplicity, and dispersal. They also share information across several agents, are self-organized, co-evolve, and learn through iterations to execute efficient search operations. Furthermore, various agents may be parallelized, making large scale optimization more viable from an implementation standpoint. Artificial bee colony optimization, ant colony optimization, Lion algorithm, whale optimization method, grasshopper optimization algorithm, Particle Swarm Optimization, Bat algorithm (BA), Stochastic diffusion search, Chaotic bat algorithm, Cat Swarm Optimization (CSO), artificial fish swarm algorithm, and elephant herding optimization, are some examples of swarm based algorithms[10].

Metaheuristic algorithms based on physics or chemistry are created differently, with inspiration drawn from known physics or chemistry occurrences. These algorithms often imitate physical or chemical laws such as electrical charges, river systems, chemical processes, gas pressure, gravity, and so on. The gravitational search algorithm created by Rashedi et al. (2009), models Newton's theory of gravitation, whereas the chemical reaction algorithm mimics chemical processes. Using control volume mass balance models, the equilibrium optimization method simulates the estimate process of equilibrium states. Magnetic charged system search, ions motion algorithm, atom search optimization, and henry gas solubility optimization are all physics/chemistry based metaheuristic algorithms[11].

III. FUNDAMENTALS OF TEACHING–LEARNING-BASED OPTIMIZATION

TLBO originates from the analogy of classroom instruction where the teacher's influence is central to improving the average performance of a class. This algorithm comprises two main phases[12]:

- Teacher Phase:

In this phase, the algorithm identifies the best solution, treating it as the "teacher." The teacher attempts to elevate the mean knowledge of the class by shifting each learner's solution toward the teacher's performance. This process is mathematically modeled by updating each candidate solution using the teacher's influence on the class average. Notably, TLBO does not require algorithm-specific parameters like learning rates or mutation rates, relying instead on basic control parameters such as population size and iteration counts[13].

- Learner Phase:

https://doi.org/10.48161/qtj.v2n2a19

After the teacher phase, individuals further improve by interacting with one another. In pairwise interactions, a learner may compare its solution with that of a peer and update its position if the peer demonstrates a superior solution. This mutual learning process, which is typically executed without additional parameter tuning, helps prevent stagnation and ensures diversity among solutions6.

The elegance of TLBO lies in its simple yet effective strategy. While many metaheuristics combine stochastic rules with parameter tuning, TLBO relies on a parameterless design that inherently balances exploration and exploitation6. The algorithm has demonstrated robust performance on both unimodal and multimodal benchmark functions, often achieving competitive or superior results compared with other state-of-the-art algorithms[14].

One limitation commonly noted is the possibility of premature convergence, especially in highly complex or combinatorial optimization landscapes, where local optima can trap the search process. To counteract this, various modifications and hybridizations have been proposed, as discussed in subsequent sections[15].

IV. TAXONOMY AND CLASSIFICATION OF TLBO VARIANTS

Over the years, numerous extensions and hybridizations of TLBO have emerged. These can be broadly categorized as follows[16]:

- Adaptive TLBO Variants:

These adaptations involve self-adaptive parameter tuning mechanisms that modulate aspects such as the teacher factor and learner selection probability dynamically during the search process. For example, some adaptive models adjust the teacher factor using a logistic growth curve to monitor convergence states.

- Multi-objective TLBO:

Designed to simultaneously tackle multiple conflicting objectives, these variants integrate mechanisms such as Pareto dominance and crowding distance to maintain a diverse set of non-dominated solutions. They are particularly useful in applications such as power loss minimization and voltage profile enhancement in distribution networks.

- Discrete and Combinatorial TLBO:

TLBO has been extended to handle discrete optimization problems like the Quadratic Assignment Problem (QAP). In these variants, TLBO is combined with recombination operators or hybridized with tools like Robust Tabu Search to manage the intricate combinatorial landscape.

- Hybrid TLBO Frameworks:

Complex optimization challenges have led to the development of hybrid frameworks where TLBO is combined with other metaheuristics. For example, the hybrid TLBO–RTS algorithm leverages TLBO's exploration capabilities with the local search strength of RTS, while methods like GA-PSO-TLBO combine the global search abilities of Genetic Algorithms and PSO with TLBO's learning mechanism.

- Intelligent and Surrogate-Assisted Extensions:

Recent work explores the integration of machine learning techniques with TLBO. Surrogate models, such as improved Radial Basis Function (RBF) networks, and dimension reduction methods like sparse autoencoders have been combined with TLBO to tackle high-dimensional, computationally expensive optimization tasks.

The following table summarizes the primary TLBO variant categories along with their key characteristics:

Table 1: TLBO variant taxonomy and characteristics.

Variant Category	**Key Features**	**Example Applications**
Adaptive TLBO	Dynamic adjustment of teacher factor, parameter-free learning rate modifications	Mechanical design optimization, portfolio design

Variant Category	Key Features	Example Applications
Multi-objective TLBO	Pareto front management, crowding distance computation, non-dominated sorting	Power distribution systems, resource allocation
Discrete and Combinatorial TLBO	Combinatorial operators, hybridization with local search (e.g., Robust Tabu Search)	Quadratic Assignment Problem (QAP)
Hybrid TLBO Frameworks	Integration with GA, PSO, and other metaheuristics for complementary strength	Engineering design, network reconfiguration
Intelligent and Surrogate-Assisted TLBO	Incorporation of machine learning models, surrogate modeling, and dimension reduction	High-dimensional optimization, cloud computing optimization

In addition to tabular representation, a conceptual flow diagram is presented below using figure 1 to illustrate the taxonomy and evolution of TLBO variants:

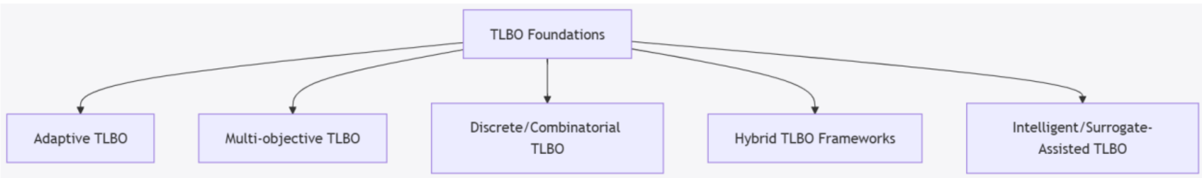

FIGURE 2. Conceptual taxonomy of TLBO variants.

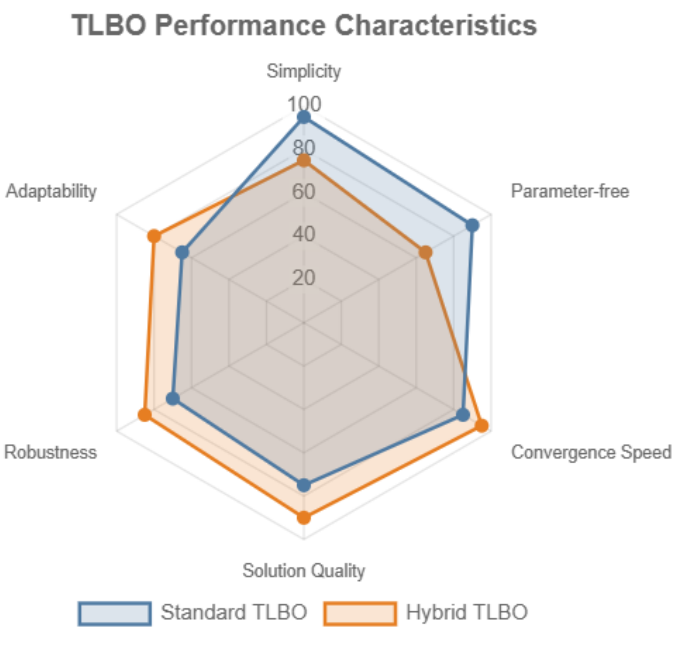

FIGURE 3. TLBO Performance Standard vs. Hybrid .

V. BENCHMARKING AND PERFORMANCE EVALUATION

A significant body of research has focused on benchmarking TLBO and its variants against established metaheuristics. Numerous test functions, including unimodal, multimodal, and composite benchmark functions, have been employed to evaluate TLBO's performance[8].

Studies have demonstrated that TLBO frequently performs favorably when compared to algorithms such as Particle Swarm Optimization (PSO)[17], Genetic Algorithms (GA)[18], and Ant Colony Optimization (ACO)[19]. In particular, TLBO's ability to deliver comparable or improved solution quality while requiring fewer parameter tunings is noteworthy. For example, in solving the Quadratic Assignment Problem (QAP), TLBO-based hybrid algorithms have been shown to surpass state-of-the-art methods in achieving both quality of solution and computational efficiency[20].

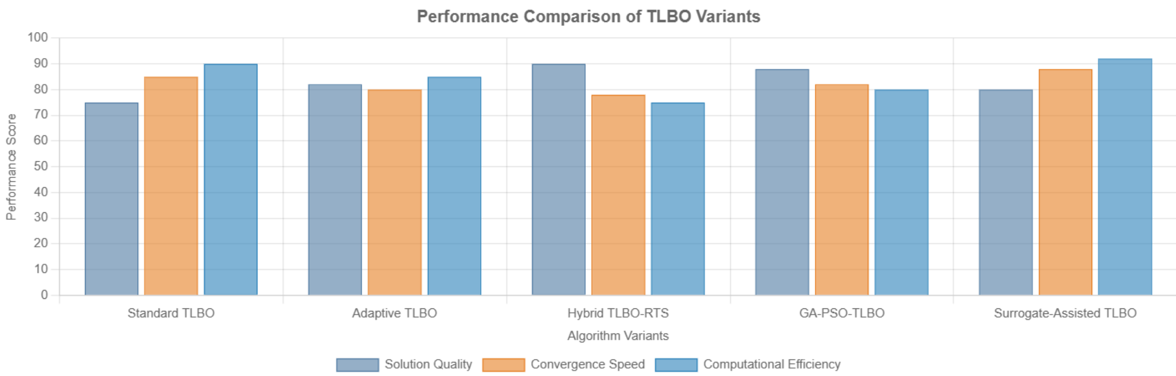

FIGURE 4. TLBO Performance comparison

The following table illustrates a comparative performance analysis based on key metrics such as best solution (BS), average solution (AS), and computational effort (CPU time):

Table 2. Comparison of TLBO variants based on benchmark performance.

Algorithm Variant	**Best Solution (BS)**	**Average Solution (AS)**	**Computational Effort (CPU Time)**	**Key Benchmark Function(s)**
Standard TLBO	High	Consistent	Low	Unimodal: F1, Multimodal: F6
Adaptive TLBO	Improved over TLBO	Better diversity	Comparable	Engineering design problems
Hybrid TLBO–RTS	Superior	High reliability	Moderate	QAP, complex combinatorial tasks
GA-PSO-TLBO	Significantly high	Superior stability	Efficient due to sub-population usage	Mechanical and gear design
Surrogate-Assisted TLBO	Competitive	Efficient in high-dimensional problems	Lower due to reduced evaluation cost	Cloud computing optimization

Several studies have also employed statistical tests to validate performance improvements. Performance metrics such as convergence speed, stability of the optimum solution across multiple runs, and robustness against premature convergence have been analyzed. The consensus among various experiments is that while standard TLBO offers fast convergence, hybrid frameworks can significantly improve exploration capability and avoid local optimum traps.

VI. APPLICATIONS OF TLBO IN COMPLEX OPTIMIZATION PROBLEMS

TLBO has been widely applied in multiple domains, demonstrating its versatility and robustness. Some notable applications include[1], [2]:

- Engineering Design:

In mechanical and structural design—such as reinforced concrete beam design, gear transmission system design, and turbine balancing—TLBO has been successfully applied to optimize cost, weight, and performance metrics while handling multiple types of design variables (continuous, discrete, and integer). TLBO's ability to incorporate learning processes without additional selection mechanisms allows it to process diverse problem structures efficiently.

- Energy Systems and Power Networks:

TLBO and its hybrid variants have been applied to complex power system reconfiguration and distribution network optimization problems. For example, hybrid TLBO–RTS algorithms have been proposed for solving the challenging Quadratic Assignment Problem (QAP), modeling real-life problems such as backboard wiring and typewriter layout design. Similarly, the hybrid TLBO–CO algorithm has shown significant improvements in power loss minimization and voltage profile enhancement in distribution networks.

- Portfolio Optimization and Financial Engineering:

Recent studies have demonstrated the use of TLBO for optimal asset allocation and portfolio selection in complex financial markets. By mapping the learning process onto economic decision-making, TLBO effectively handles multi-objective trade-offs, balancing risk and return without the need for fine-tuning of control parameters.

- Cloud and Edge Computing:

In high-dimensional optimization scenarios such as computation offloading in mobile edge computing, surrogate-assisted TLBO algorithms have been developed. These methods integrate sparse autoencoders and improved Radial Basis Function (RBF) models to reduce the dimensionality of the search space and the computational cost, while maintaining robustness in obtaining near-optimal solutions.

- Scheduling and Resource Allocation:

TLBO-based approaches have also been applied to scheduling problems and resource allocation in manufacturing systems. Its two-phase design lends itself well to balancing global search with local refinement, thereby enhancing the quality of scheduling outcomes under uncertainty.

The diverse range of applications underscores TLBO's potential for widespread use and highlights the ongoing interest in extending and hybridizing this method to address ever more complex optimization challenges.

VII. HYBRID AND INTELLIGENT TLBO FRAMEWORKS

As the complexity of optimization problems has grown, researchers have increasingly turned to hybrid frameworks that integrate TLBO with complementary metaheuristic algorithms. Such hybridizations seek to merge the global exploration strengths of one algorithm with the rapid convergence capabilities of TLBO. Some prominent hybrid frameworks include[21], [22], [23], [24]:

- Hybrid TLBO–RTS (Robust Tabu Search):

In this approach, TLBO is used to generate a diverse population of candidate solutions, after which a Robust Tabu Search (RTS) engine refines these candidates through local search to escape local optima. The RTS component, which benefits from its ability to explore unexplored areas by preventing reverse moves using a tabu list, helps TLBO overcome pitfalls of premature convergence.

- Parallel and Cooperative Hybrid Frameworks (hTLBO–CO):

The hTLBO–CO algorithm introduces a parallel-cooperative hybridization architecture where the TLBO Teacher Phase, the Cheetah Optimizer (CO) Hunting Phase, and the TLBO Learner Phase operate concurrently. A shared elite pool facilitates continuous information exchange, ensuring a dynamic balance between exploration and exploitation. This approach has demonstrated significant improvements in voltage profile enhancement and power loss minimization in reconfigured distribution networks.

- GA-PSO-TLBO Hybrid Methods:

The GA-PSO-TLBO framework has been designed to simultaneously leverage the global search capabilities of Genetic Algorithms (GA), the local search efficiency of Particle Swarm Optimization (PSO), and the learning strength of TLBO. In these hybrid algorithms, sub-populations are strategically assigned to different search loops. For instance, a sub-population with superior fitness values might be processed using GA and TLBO loops, while a sub-population with inferior fitness values is refined using PSO, thereby reducing search time and accelerating convergence.

- Intelligent TLBO with Surrogate Models and Dimension Reduction:

High-dimensional optimization problems often present excessive computational demands. Recently, TLBO has been hybridized with machine learning techniques such as sparse autoencoders and surrogate models like improved Radial Basis Function (RBF) networks. In such frameworks, the high-dimensional search space is compressed into a lower-dimensional manifold. The surrogate model then approximates the objective function, thus reducing the number of costly true evaluations. This innovation enables efficient optimization in areas like cloud computing and resource scheduling.

Below is a flowchart that summarizes the structure of a typical hybrid TLBO framework, highlighting the integration of TLBO with RTS, GA, and PSO components[25], [26]:

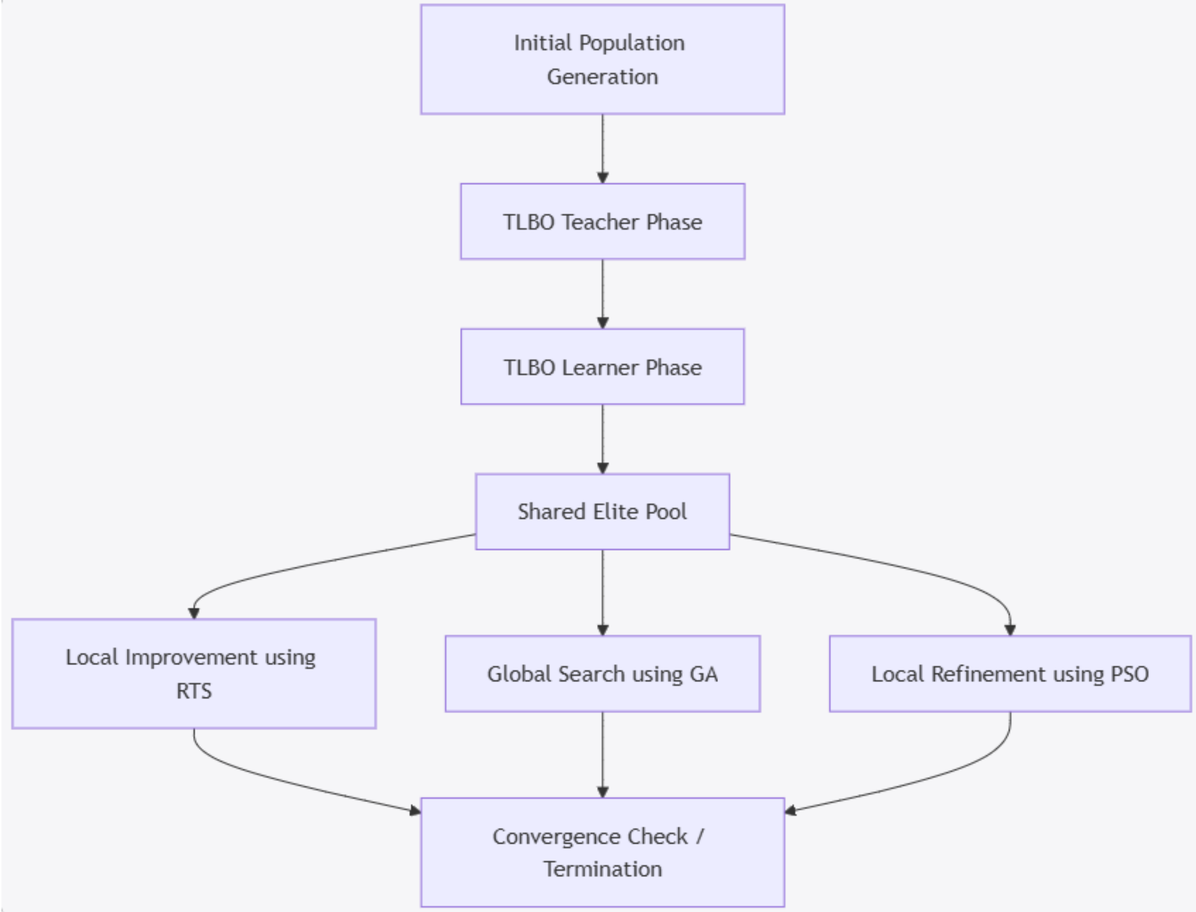

FIGURE 5. Flowchart of a hybrid TLBO framework Integrating RTS, GA, and PSO components

The integration of these complementary algorithms not only significantly enhances convergence speed but also improves the quality of the final solution by efficiently escaping local optima. The self-adaptive mechanisms embedded in hybrid schemes further adjust the relevance of each component based on real-time feedback from the optimization process.

VIII. THEORETICAL ANALYSIS AND CONVERGENCE PROPERTIES

Understanding the theoretical underpinnings of TLBO and its variants is vital for assessing their robustness and efficiency. Although TLBO is primarily heuristic in nature, several studies have provided insights into its mathematical behavior and convergence properties[23].

1. CONVERGENCE PROPERTIES

TLBO can be described in the context of Markov chains, where each iteration represents a state transition influenced by teacher and learner interactions. Under certain mild conditions and by maintaining a non-zero probability of exploration, it has been shown that the probability of the algorithm converging to the set of global optima approaches one as the number of iterations increases. This guarantee of asymptotic convergence offers theoretical reassurance of TLBO's long-term optimization potential[24].

Mathematically, the update equations in both the teacher phase and the learner phase are designed to push the population toward improved fitness values. For example, during the teacher phase, the transformation of solutions is formulated as:

$$X_{\text{new}} = X_{\text{old}} + r\,(X_{\text{teacher}} - T_F \times M)$$

where r is a random number, T_F is the teaching factor (which may adapt according to logistic growth curves), and M denotes the mean of the class. Similar iterative update rules in the learner phase further refine the search by leveraging pairwise comparisons. Such strategies ensure that, despite the absence of explicit mutation or crossover operators found in GA, the population gradually converges toward optimal regions6.

2. COMPUTATIONAL COMPLEXITY

The per-iteration computational complexity of TLBO is generally linear with respect to both the population size N and the problem dimensionality D. This efficiency arises from TLBO's avoidance of elaborate parameter tuning and selection mechanisms. Although in some hybrid formulations additional overhead is introduced (e.g., from surrogate model training or local search engines such as RTS), the overall complexity remains competitive when compared with other metaheuristic approaches such as PSO or GA6.

3. STABILITY AND ROBUSTNESS

TLBO exhibits strong stability across different optimization problems. Its parameter-free nature reduces the risk of divergent behavior due to improper parameter settings. However, in highly complex or multimodal landscapes, complementary strategies, such as adaptive or hybrid TLBO variants, become necessary to maintain diversity and avoid premature convergence56.

IX. CHALLENGES AND LIMITATIONS

Despite its many strengths, TLBO and its variants face several challenges that researchers are actively addressing[21], [27], [28]:

- Premature Convergence:

TLBO, particularly in its standard form, may sometimes converge prematurely in complex or high-dimensional optimization landscapes, leading to suboptimal solutions. Hybrid approaches seek to resolve this issue, but the challenge remains critical in unconstrained environments56.

- Scalability to High-Dimensional Problems:

While TLBO shows robustness in many settings, scaling to very high-dimensional problems may require integration with dimension reduction techniques or surrogate modeling to manage computational cost effectively4.

- Limited Control Over Exploration–Exploitation Balance:

Although TLBO is designed to balance exploration and exploitation through its two-phase approach, different problem characteristics may demand more nuanced control strategies. This gap has motivated the development of adaptive variants that monitor and adjust internal parameters dynamically6.

- Application-Specific Customizations:

In problems where constraints and domain-specific knowledge are critical (e.g., engineering design or power network reconfiguration), standard TLBO must be modified or hybridized with specialized operators to ensure feasibility and efficiency[29], [30].

X. FUTURE RESEARCH DIRECTIONS

Based on the challenges identified, several future research avenues emerge:

- Development of Advanced Hybrid Models:

Future work should focus on more sophisticated hybridization schemes that integrate TLBO with both global search methods (e.g., GA) and local search enhancements (e.g., RTS) in a parallel and cooperative framework. Such models can dynamically allocate computational resources and further delay the onset of premature convergence.

- Integration with Machine Learning:

Incorporating intelligent learning techniques such as deep neural networks, sparse autoencoders, and surrogate modeling could streamline TLBO for high-dimensional and computationally expensive problems. This integration could lead to algorithms that adaptively reduce dimensionality and predict objective functions more efficiently.

- Multi-objective and Constraint-Handling Extensions:

Expanding TLBO to handle multi-objective optimization and complex constraints in real-world scenarios remains a vibrant research direction. Developing robust frameworks that can manage diverse constraints while maintaining population diversity could elevate TLBO to a premier tool for industrial applications.

- Theoretical Foundation and Convergence Analysis:

Although some theoretical analyses exist, a deeper mathematical understanding of TLBO, especially in hybrid and adaptive forms, is essential. Future research should aim at formal proofs, tighter convergence bounds, and comprehensive analyses of algorithmic behavior under different assumptions.

- Real-World Application and Benchmarking:

Extensive benchmarking on large-scale, industry-relevant problems is necessary to validate TLBO's practical utility. Continued collaboration between academia and industry can help refine these methods to meet real-world demands.

XI. CONCLUSION

In this comprehensive review, we have examined the evolution and hybridization of Teaching–Learning-Based Optimization over recent years. Key contributions of TLBO include its simple, parameter-free design and effective two-phase learning mechanism that mimics the natural process of teacher–student interaction. The inherent advantages of TLBO have led to numerous extensions and hybrid frameworks that integrate it with other metaheuristics such as GA, PSO, RTS, and intelligent surrogate models.

Below is a bullet-list summary of the main findings:

- Simplicity and Robustness: TLBO's parameterless design and teacher–learner structure enable efficient exploration and quick convergence while reducing the need for parameter tuning6.
- Hybrid Enhancements: Hybrid frameworks like TLBO–RTS, hTLBO–CO, and GA-PSO-TLBO demonstrate improved performance by combining global search and local refinement techniques, significantly overcoming premature convergence and enhancing solution quality.
- Diverse Applications: TLBO has been successfully applied in mechanical design, power systems, portfolio optimization, and cloud computing, among other fields. Its adaptability makes it suitable for a wide range of optimization tasks.
- Theoretical and Computational Efficiency: The algorithm's linear per-iteration complexity with respect to population size and dimensionality, coupled with convergence guarantees under certain conditions, underscores its computational efficiency.
- Future Prospects: Challenges remain in scaling to high-dimensional problems, managing exploration–exploitation trade-offs, and handling complex constraints. Future work in advanced hybridization and intelligent integration compared to conventional approaches is promising.

In summary, TLBO has evolved from a novel, classroom-inspired algorithm to a versatile and robust metaheuristic tool that continues to inspire innovative research. Its promising performance across an array of applications, coupled with ongoing advancements in hybrid and adaptive variants, ensures that TLBO will remain a pillar in the field of metaheuristic optimization for years to come.

The critical insights and extensive developments presented in this review offer a roadmap for both academic research and practical applications. By combining rigorous theoretical analysis with empirical benchmarking and real-world applications, it is evident that the hybrid evolution of TLBO enhances not only the algorithm's performance but also its flexibility in tackling increasingly complex optimization challenges.

REFERENCES

[1] R. V. Rao, "Teaching-Learning-Based Optimization Algorithm," in *Teaching Learning Based Optimization Algorithm*, Cham: Springer International Publishing, 2016, pp. 9–39. doi: 10.1007/978-3-319-22732-0_2.

[2] F. Zou, D. Chen, and Q. Xu, "A survey of teaching–learning-based optimization," *Neurocomputing*, vol. 335, pp. 366–383, Mar. 2019, doi: 10.1016/j.neucom.2018.06.076.

[3] S. M. Almufti and A. A. Shaban, "U-Turning Ant Colony Algorithm for Solving Symmetric Traveling Salesman Problem," *Academic Journal of Nawroz University*, vol. 7, no. 4, p. 45, Dec. 2018, doi: 10.25007/ajnu.v7n4a270.

[4] S. M. Almufti, R. R. Asaad, and B. W. Salim, "Review on Elephant Herding Optimization Algorithm Performance in Solving Optimization Problems," *Article in International Journal of Engineering and Technology*, vol. 7, no. 4, pp. 6109–6114, 2018, doi: 10.14419/ijet.v7i4.23127.

[5] 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.
[6] M. Črepinšek, S.-H. Liu, and L. Mernik, "A note on teaching–learning-based optimization algorithm," *Inf Sci (N Y)*, vol. 212, pp. 79–93, Dec. 2012, doi: 10.1016/j.ins.2012.05.009.
[7] S. Mohammed Almufti, R. P. Maribojoc, and A. V. Pahuriray, "Ant Based System: Overview, Modifications and Applications from 1992 to 2022," *Polaris Global Journal of Scholarly Research and Trends*, vol. 1, no. 1, pp. 29–37, Oct. 2022, doi: 10.58429/pgjsrt.v1n1a85.
[8] 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.
[9] S. M. Almufti, R. Boya Marqas, and V. Ashqi Saeed, "Taxonomy of bio-inspired optimization algorithms," *Journal of Advanced Computer Science & Technology*, vol. 8, no. 2, p. 23, Aug. 2019, doi: 10.14419/jacst.v8i2.29402.
[10] A. Kaveh, S. R. Hoseini Vaez, and P. Hosseini, "MATLAB CODE FOR AN ENHANCED VIBRATING PARTICLES SYSTEM ALGORITHM," 2018.
[11] S. M. Almufti, "Artificial Bee Colony Algorithm performances in solving Welded Beam Design problem," *Computer Integrated Manufacturing Systems*, vol. 28, 2022, doi: 10.24297/j.cims.2022.12.17.
[12] K. Y. Gómez Díaz, S. E. De León Aldaco, J. Aguayo Alquicira, M. Ponce-Silva, and V. H. Olivares Peregrino, "Teaching–Learning-Based Optimization Algorithm Applied in Electronic Engineering: A Survey," *Electronics (Basel)*, vol. 11, no. 21, p. 3451, Oct. 2022, doi: 10.3390/electronics11213451.
[13] D. Chen, F. Zou, Z. Li, J. Wang, and S. Li, "An improved teaching–learning-based optimization algorithm for solving global optimization problem," *Inf Sci (N Y)*, vol. 297, pp. 171–190, Mar. 2015, doi: 10.1016/j.ins.2014.11.001.
[14] R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems," *Computer-Aided Design*, vol. 43, no. 3, pp. 303–315, Mar. 2011, doi: 10.1016/j.cad.2010.12.015.
[15] R. V. Rao, *Teaching Learning Based Optimization Algorithm*. Cham: Springer International Publishing, 2016. doi: 10.1007/978-3-319-22732-0.
[16] F. Zou, L. Wang, X. Hei, and D. Chen, "Teaching–learning-based optimization with learning experience of other learners and its application," *Appl Soft Comput*, vol. 37, pp. 725–736, Dec. 2015, doi: 10.1016/j.asoc.2015.08.047.
[17] 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.
[18] 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.
[19] S. M. Almufti, "U-Turning Ant Colony Algorithm powered by Great Deluge Algorithm for the solution of TSP Problem," 2015.
[20] 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.
[21] D. Wu, S. Wang, Q. Liu, L. Abualigah, and H. Jia, "An Improved Teaching-Learning-Based Optimization Algorithm with Reinforcement Learning Strategy for Solving Optimization Problems," *Comput Intell Neurosci*, vol. 2022, 2022, doi: 10.1155/2022/1535957.
[22] A. K. Shukla, P. Singh, and M. Vardhan, "An adaptive inertia weight teaching-learning-based optimization algorithm and its applications," *Appl Math Model*, vol. 77, 2020, doi: 10.1016/j.apm.2019.07.046.
[23] Y. Ma, X. Zhang, J. Song, and L. Chen, "A modified teaching–learning-based optimization algorithm for solving optimization problem," *Knowl Based Syst*, vol. 212, 2021, doi: 10.1016/j.knosys.2020.106599.
[24] C. Wu, J. Zhao, Y. Feng, and M. Lee, "'Solving discounted {0-1} knapsack problems by a discrete hybrid teaching-learning-based optimization algorithm,'" *Applied Intelligence*, vol. 50, no. 6, 2020, doi: 10.1007/s10489-020-01652-0.
[25] B. Crawford *et al.*, "A teaching-learning-based optimization algorithm for the weighted set-covering problem," *Tehnicki Vjesnik*, vol. 27, no. 5, 2020, doi: 10.17559/TV-20180501230511.
[26] E. Naderi, M. Pourakbari-Kasmaei, and M. Lehtonen, "Transmission expansion planning integrated with wind farms: A review, comparative study, and a novel profound search approach," *International Journal of Electrical Power and Energy Systems*, vol. 115, 2020, doi: 10.1016/j.ijepes.2019.105460.
[27] F. A. Zeidabadi, M. Dehghani, P. Trojovský, Š. Hubálovský, V. Leiva, and G. Dhiman, "Archery Algorithm: A Novel Stochastic Optimization Algorithm for Solving Optimization Problems," *Computers, Materials and Continua*, vol. 72, no. 1, 2022, doi: 10.32604/cmc.2022.024736.
[28] G. Zhou, Y. Zhou, W. Deng, S. Yin, and Y. Zhang, "Advances in teaching–learning-based optimization algorithm: A comprehensive survey(ICIC2022)," *Neurocomputing*, vol. 561, 2023, doi: 10.1016/j.neucom.2023.126898.
[29] 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.

[30] S. M. Almufti, “Fusion of Water Evaporation Optimization and Great Deluge: A Dynamic Approach for Benchmark Function Solving,” *Fusion: Practice and Applications*, vol. 13, no. 1, pp. 19–36, 2023, doi: 10.54216/FPA.130102.