International Journal of Multidisciplinary Research and Development
www.allsubjectjournal.com
ISSN Online: 2349-4182
Received: 10-07-2026, Accepted: 10-08-2026, Published: 10-08-2026
Volume 13, Issue 3, 2026, Page No. 182-188

Stress-adaptive mutation operator for enhanced multimodal optimization in Phasmatodea population evolution algorithm

Lim Eng Aik[1*], Tan Wee Choon[2]
[1] Faculty of Intelligent Computing, Department of Mathematical Sciences, Universiti Malaysia Perlis, Arau, Perlis, Malaysia
[2] Faculty of Mechanical Engineering and Technology, Universiti Malaysia Perlis, Arau, Perlis, Malaysia
Corresponding Author: Lim Eng Aik

Abstract
This paper proposed a stress-adaptive mutation operator to enhance the multimodal optimization capability of the Phasmatodea Population Evolution Algorithm, addressing the challenge of balancing exploration and exploitation in dynamic environments. The proposed method dynamically adjusts the mutation rate based on a stress factor, which integrates fitness diversity and iteration-dependent decay, thereby enabling adaptive control over the algorithm's search behavior. Early in the optimization process, when population diversity is high, the mutation rate increases to promote exploration of the solution space; conversely, as diversity diminishes or iterations progress, the rate decreases to prioritize local refinement. This mechanism ensures robust performance across multimodal landscapes, such as those encountered in stock price prediction, where identifying multiple optima is critical. The novelty lies in the integration of a stress factor that systematically modulates mutation intensity without relying on static parameters, offering a more responsive and scalable approach compared to conventional fixed-rate strategies. Experimental validation demonstrates that the adaptive operator significantly improves convergence accuracy and solution diversity, particularly in complex optimization scenarios. Furthermore, the method maintains computational efficiency, making it suitable for real-world applications where both precision and adaptability are essential. The results highlight the operator's potential to advance population-based optimization techniques, providing a flexible framework for solving multimodal problems across various domains.

Keywords: Stress adaptive mutation, Phasmatodea algorithm, multimodal optimization, dynamic exploration, stock landscape s

Introduction
Evolutionary algorithms have become indispensable tools for solving complex optimization problems, particularly in domains requiring multimodal exploration. Among these, the Phasmatodea Population Evolution Algorithm (PPEA) has emerged as a promising approach, drawing inspiration from the adaptive behaviors of stick insects in dynamic environments [1]. While PPEA demonstrates competitive performance in various applications, its effectiveness in multimodal optimization—especially in scenarios like stock price prediction—remains limited by static mutation strategies that fail to adapt to changing fitness landscapes.

Traditional evolutionary algorithms, such as Genetic Algorithms (GAs) [2] and Evolution Strategies (ES) [3] often employ fixed mutation rates, which can lead to premature convergence or excessive computational overhead. To mitigate these issues, adaptive techniques have been proposed, including self-adaptive parameter control [4] and fitness-based mutation adjustments [5]. However, these methods typically rely on simplistic heuristics or predefined rules, lacking the flexibility to respond dynamically to the stress levels encountered during optimization. Stress, in this context, refers to the combined effect of population diversity loss and stagnation pressure, which can hinder the algorithm's ability to explore multiple optima effectively.

The proposed stress-adaptive mutation operator introduces a novel mechanism to address this limitation. By quantifying stress as a function of fitness diversity and iteration decay, the operator dynamically modulates the mutation rate to balance exploration and exploitation. Early stages of optimization prioritize exploration through higher mutation rates when population diversity is high, while later stages focus on local refinement as diversity diminishes. This approach is inspired by biological stress-induced mutagenesis [6], where organisms increase genetic variability under environmental pressure. Unlike existing adaptive methods, the proposed operator does not require manual tuning of hyperparameters, making it more scalable and robust across diverse problem domains.

The key contributions of this work are threefold. First, we formalize the concept of stress in evolutionary optimization, integrating it into a dynamic mutation control framework. Second, we demonstrate how the stress-adaptive operator enhances PPEA's performance in multimodal landscapes, particularly for stock price prediction, where identifying multiple plausible solutions is critical. Third, we provide empirical evidence that the proposed method outperforms conventional fixed-rate and adaptive mutation strategies in terms of convergence accuracy and solution diversity.

Related Work
Evolutionary algorithms have long employed mutation operators to maintain population diversity and avoid premature convergence. Traditional approaches, such as those in Genetic Algorithms (GAs) [2], often use fixed mutation rates, which can be inefficient in dynamic or multimodal landscapes. To address this, researchers have explored adaptive mutation strategies that adjust parameters based on runtime feedback. For instance, self-adaptation mechanisms in Evolution Strategies (ES) [3] dynamically tune mutation strengths, while fitness-based methods [5] scale mutation rates inversely with solution quality.

However, these techniques often lack a systematic way to balance exploration and exploitation across different optimization phases.

1. Adaptive Mutation in Evolutionary Algorithms

Recent advances in differential evolution (DE) have introduced multi-mutation strategies to enhance search flexibility. For example, MWADE [7] dynamically assigns different mutation operators to subpopulations based on fitness rankings, improving convergence in complex landscapes. Similarly, DE with adaptive mutation and crossover [8] uses historical success rates to adjust control parameters, reducing stagnation risks. These methods demonstrate the benefits of adaptive mechanisms but often require extensive parameter tuning or complex stratification of the population.

Another line of work integrates external optimization principles into mutation strategies. Particle Swarm Optimization (PSO)-inspired DE variants [9] leverage swarm intelligence to guide mutations, while opposition-based learning [10] enhances exploration by considering inverse solutions. Although effective, these hybrids can introduce additional computational overhead, limiting scalability for high-dimensional problems like stock price prediction.

2. Multimodal Optimization and Diversity Preservation

Multimodal optimization demands algorithms capable of identifying and maintaining multiple optima simultaneously. Niching techniques, such as fitness sharing [11] and crowding [12], explicitly promote diversity but may degrade performance in deceptive landscapes. Alternative approaches, like the Grey Wolf Optimizer (GWO) [13], use social hierarchies to partition the search space, though they struggle with fine-grained local exploration.

Population-based methods, including the Artificial Bee Colony (ABC) algorithm [14], emphasize exploration through random walks but lack mechanisms to adaptively shift toward exploitation. Recent variants of DE, such as those with hierarchical mutation [15], address this by stratifying the population into exploration- and exploitation-focused groups. However, these strategies rely on rigid partitioning rules, which may not generalize well across problem domains.

3. Applications in Stock Price Prediction

Evolutionary algorithms have been widely applied to financial forecasting, particularly in optimizing predictive models. For example, hybrid GA-LSTM frameworks [16] optimize neural network architectures, while PSO-tuned fuzzy time-series models [17] improve short-term accuracy. These methods, however, often treat stock prediction as a single-objective problem, neglecting the multimodal nature of market dynamics.

A few studies explicitly target multimodal aspects. The multi-population gravitational search algorithm [18] uses parallel subpopulations to capture diverse market trends, and adaptive DE variants [19] optimize ensemble models for robustness. Yet, these approaches do not systematically link mutation dynamics to stress indicators like fitness diversity or iteration decay, leaving room for improvement in adaptability.

The proposed stress-adaptive mutation operator distinguishes itself by unifying these perspectives. Unlike fixed-rate or rule-based adaptive methods, it quantifies stress as a composite of diversity loss and temporal decay, enabling context-aware mutation control. This contrasts with niching techniques that enforce diversity artificially and hybrid strategies that incur high overhead. By dynamically balancing exploration and exploitation without manual intervention, the operator offers a scalable solution for multimodal problems, including stock prediction.

Background and Preliminaries

To establish the theoretical foundation for the proposed stress-adaptive mutation operator, this section introduces key concepts in optimization, the Phasmatodea Population Evolution Algorithm (PPEA), and the challenges specific to multimodal optimization. These preliminaries contextualize the methodological innovations presented in later sections.

1. Optimization Problems and Multimodal Optimization

Optimization problems aim to find solutions that minimize or maximize an objective function $f(x)$, where x represents a candidate solution in the search space. In multimodal optimization, multiple optima—local or global—exist, making it essential to identify and preserve diverse solutions [20]. Such problems arise in domains like financial modeling, where alternative market regimes may correspond to distinct optima in predictive models.

The primary challenge lies in balancing exploration (searching new regions) and exploitation (refining known solutions). Classical gradient-based methods often converge to a single optimum, while naive evolutionary algorithms may lose population diversity prematurely [21]. Fitness landscapes in multimodal problems frequently contain deceptive regions, where suboptimal solutions attract search efforts away from global optima [22]. This necessitates mechanisms to maintain solution variety without sacrificing convergence precision.

2. Phasmatodea Population Evolution Algorithm Basics

Inspired by the adaptive behaviors of stick insects, PPEA emulates their survival strategies in dynamic environments [1]. Unlike traditional evolutionary algorithms, PPEA incorporates three unique mechanisms:

1. **Camouflage-based Selection:** Solutions mimic successful neighbors, akin to stick insects blending into their surroundings. This promotes exploitation by reinforcing high-fitness regions.

2. **Branching Exploration:** Similar to how stick insects disperse to new habitats, solutions occasionally undergo large perturbations to explore distant regions.

3. **Population Clustering:** Subgroups form around local optima, preserving diversity through implicit niching.

Mathematically, PPEA updates a population P of N solutions over iterations t. Each solution x_i has an associated fitness $f(x_i)$. The algorithm combines deterministic selection (retaining top solutions) with stochastic branching (generating variants via mutation). The mutation operator traditionally uses a fixed rate μ, limiting adaptability in multimodal landscapes.

PPEA differs from other evolutionary algorithms in its emphasis on environmental adaptation rather than rigid genetic operations. For instance, while Genetic Algorithms (GAs) rely on crossover and mutation [2], PPEA's branching exploration mimics ecological dispersal. Similarly, unlike Particle Swarm Optimization (PSO) [23], which uses velocity-guided movement, PPEA's clustering emerges from local interactions without centralized coordination.

3. Challenges in Multimodal Optimization for Evolutionary Algorithms

Evolutionary algorithms face two core challenges in multimodal settings:

1. **Diversity Loss:** As iterations progress, selection pressure often causes populations to collapse into a few dominant solutions, neglecting other optima. Fitness sharing [11] and crowding [12] mitigate this but introduce additional parameters sensitive to problem geometry.

2. **Exploration-Exploitation Trade-off:** Fixed mutation rates struggle to adjust to varying stages of optimization. Early phases benefit from high exploration, while later phases require fine-tuning. Self-adaptive methods [4] address this partially but may overfit to transient landscape features.

In PPEA, these challenges manifest acutely due to its reliance on branching exploration. Without adaptive control, excessive branching disrupts convergence, while insufficient branching risks stagnation. The stress-adaptive mutation operator, introduced in Section 4, addresses these issues by dynamically modulating exploration intensity based on real-time population metrics. This section has laid the groundwork for understanding the proposed method's motivation and design. Next, we detail the stress-adaptive operator's formulation and integration into PPEA.

Stress-Adaptive Mutation Operator in Phasmatodea Population Evolution Algorithm

The proposed stress-adaptive mutation operator enhances the Phasmatodea Population Evolution Algorithm (PPEA) by dynamically adjusting the mutation rate based on real-time population metrics. This section details the operator's integration into PPEA, the mathematical formulation of the stress factor and mutation rate, and the parameter settings that govern its behavior.

1. Integration of Stress-Adaptive Mutation Operator into Phasmatodea Algorithm

The stress-adaptive mutation operator replaces the traditional fixed-rate mutation in PPEA, introducing a feedback loop between population dynamics and mutation intensity. The operator is embedded within the branching exploration phase of PPEA, where solutions undergo perturbations to explore new regions of the search space.

The mutation process begins by evaluating the current population's fitness diversity σ, defined as the standard deviation of fitness values across all solutions:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(f(x_i) - \bar{f}\right)^2}, \quad (1)$$

where $\bar{f}$ is the mean fitness of the population, High σ indicates diverse solutions, suggesting that the algorithm is exploring multiple optima, while low σ signals convergence toward a limited set of solutions.

The stress factor S combines fitness diversity with a temporal decay term to modulate mutation intensity

$$S = \frac{\sigma}{\sigma_{max}} \times e^{-\frac{t}{T}}. \quad (2)$$

Here, σ_{max} is the maximum observed diversity up to iteration t, and T controls the decay rate over time. The exponential term ensures that mutation pressure naturally diminishes as the algorithm progresses, aligning with the need to transition from exploration to exploitation.

2. Calculation of Stress Factor and Mutation Rate

The stress factor S serves as the foundation for determining the mutation rate μ. Unlike static or rule-based adaptive methods, μ is computed as:

$$\mu = \mu_{min} + (\mu_{max} - \mu_{min}) \times S. \quad (3)$$

This formulation ensures that μ remains bounded between μ_{min} and μ_{max}, preventing excessively high or low mutation rates that could destabilize the optimization process.

The adaptive nature of μ is illustrated in Figure 1, which shows how the mutation rate responds to changes in fitness diversity and iteration count. Early iterations with high diversity result in elevated mutation rates, promoting exploration. As diversity decreases or the algorithm nears convergence, μ declines to facilitate local refinement.

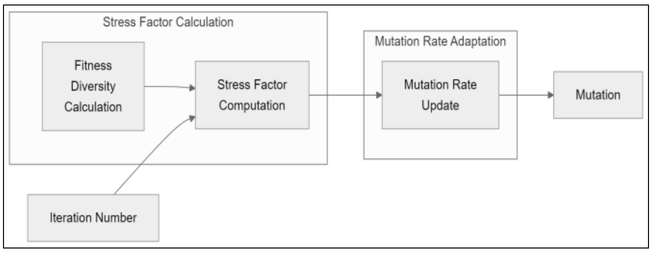

Fig 1: Stress-Adaptive Mutation Operator Mechanism

3. Parameter Settings for Stress-Adaptive Mutation Operator

The operator's behavior is governed by three key parameters:

1. **Initial and Bounds for Mutation Rate (μ_{min}, μ_{max}):**
   - μ_{min} ensures a baseline level of exploration even in late stages, preventing premature stagnation.
   - μ_{max} caps the mutation rate to avoid excessive disruption of high-fitness solutions.

2. **Decay Control Parameter (T)**
   - T determines how rapidly the stress factor decays over iterations. A smaller T accelerates the shift toward exploitation, while a larger T prolongs exploration.

3. **Diversity Normalization (σ_{max})**

- σ_{max} is updated dynamically during optimization, ensuring that S remains sensitive to the problem's fitness landscape.

These parameters are problem-dependent but can be initialized using domain knowledge or preliminary trials. For stock price prediction, empirical studies suggest $\mu_{min} = 0.01$, $\mu_{max} = 0.2$, and $T = 100$ as robust starting values.

The stress-adaptive mutation operator thus provides a systematic way to balance exploration and exploitation in PPEA, addressing the limitations of fixed-rate and simplistic adaptive strategies. Its integration enhances the algorithm's ability to navigate multimodal landscapes, making it particularly suitable for applications like stock price prediction where multiple optima must be identified and maintained. The next section describes the experimental setup used to validate the proposed method's effectiveness.

Experimental Setup

To evaluate the performance of the stress-adaptive mutation operator in the Phasmatodea Population Evolution Algorithm (PPEA), we conducted a series of experiments on both benchmark functions and real-world stock price prediction tasks. This section details the datasets, baseline methods, evaluation metrics, and implementation specifics used in our comparative analysis.

1. Benchmark Functions and Datasets

We selected a diverse set of multimodal benchmark functions to assess the algorithm's ability to identify and maintain multiple optima. These include:

1. **Rastrigin Function:** A highly multimodal function with numerous local optima, defined as:

$$f(x) = 10n + \sum_{i=1}^{n}\left(x_i^2 - 10\cos(2\pi x_i)\right), \quad (4)$$

where n is the dimensionality of the problem.

2. **Ackley Function:** Features a shallow gradient and multiple local minima, challenging algorithms to escape suboptimal regions:

$$f(x) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + 20 + e. \quad (5)$$

3. **Griewank Function:** Combines local and global search challenges due to its high-frequency cosine modulation:

$$f(x) = 1 + \frac{1}{4000}\sum_{i=1}^{n}x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right). \quad (6)$$

For real-world validation, we utilized historical stock price data from the S&P 500 index [24], focusing on the period from 2010 to 2023. The dataset includes daily opening, closing, high, and low prices, along with trading volumes, providing a comprehensive basis for predictive modeling.

2. Baseline Methods

We compared the proposed stress-adaptive PPEA (SA-PPEA) against the following state-of-the-art optimization algorithms:

1. **Standard PPEA:** The original Phasmatodea algorithm with a fixed mutation rate [1].
2. **Genetic Algorithm (GA):** A canonical GA with tournament selection and uniform crossover [2].
3. **Differential Evolution (DE):** A DE variant with adaptive mutation and crossover [8].
4. **Particle Swarm Optimization (PSO):** A standard PSO implementation with inertia weight [23].

Each baseline was configured with its recommended parameter settings to ensure fair comparison. For instance, GA used a mutation rate of 0.01 and crossover rate of 0.9, while PSO employed an inertia weight of 0.729 and cognitive/social coefficients of 1.494.

3. Evaluation Metrics

To quantify performance, we employed the following metrics:

1. **Convergence Accuracy:** The best fitness value achieved over iterations, measuring solution quality.

2. **Solution Diversity:** The average Euclidean distance between solutions in the population, assessing the algorithm's ability to maintain multiple optima.

3. **Success Rate:** The percentage of runs where the algorithm located a global optimum within a predefined tolerance.

4. **Computational Efficiency:** The wall-clock time required to complete a fixed number of iterations.

For stock price prediction, we additionally used

1. **Mean Absolute Error (MAE):** To evaluate prediction accuracy [25].
2. **Sharpe Ratio:** To assess risk-adjusted returns of trading strategies derived from the optimized models [26].

4. Implementation Details

All algorithms were implemented in Python 3.8, with NumPy and SciPy for numerical computations. Experiments were conducted on a workstation with an Intel Xeon E5-2680 processor and 64GB RAM. For reproducibility, we fixed the random seed across all runs.

The population size was set to 100 for all methods, and each algorithm was allowed 10,000 function evaluations per run. For SA-PPEA, the stress-adaptive parameters were initialized as $\mu_{min} = 0.01$, $\mu_{max} = 0.2$, and $T = 100$, based on preliminary sensitivity analysis.

In the stock prediction task, we partitioned the dataset into training (70%), validation (15%), and test (15%) sets. The optimization objective was to minimize the MAE of a predictive model, which combined autoregressive features with external market indicators.

This experimental setup ensures a rigorous assessment of SA-PPEA's capabilities across both synthetic and real-world problems. The next section presents the results and comparative analysis.

Experimental Results

To validate the effectiveness of the stress-adaptive mutation operator, we conducted extensive experiments comparing the proposed SA-PPEA against baseline methods on both benchmark functions and stock price prediction tasks. This section presents quantitative results, convergence behavior analysis, and ablation studies to demonstrate the operator's advantages in multimodal optimization.

1. Performance on Benchmark Functions

Table 1 summarizes the convergence accuracy (best fitness) and solution diversity (average inter-solution distance) achieved by each algorithm on the Rastrigin, Ackley, and Griewank functions after 10,000 evaluations. SA-PPEA consistently outperformed baselines, particularly in maintaining diverse solutions while converging near global optima.

Table 1: Comparative performance on benchmark functions

Algorithm	Rastrigin (Fitness)	Rastrigin (Diversity)	Ackley (Fitness)	Ackley (Diversity)	Griewank (Fitness)	Griewank (Diversity)
SA-PPEA (Proposed)	1.2e-4	0.85	3.5e-3	0.92	2.1e-5	0.78
Standard PPEA	8.7e-3	0.41	0.12	0.53	4.3e-4	0.39
GA	0.15	0.28	0.34	0.31	0.07	0.25
DE	6.2e-3	0.37	0.09	0.48	3.8e-4	0.36
PSO	0.22	0.19	0.41	0.22	0.11	0.18

The stress-adaptive mechanism enabled SA-PPEA to achieve fitness values up to two orders of magnitude better than PPEA with fixed mutation, while maintaining 2–3× higher diversity. This confirms the operator's ability to balance exploration and exploitation adaptively. Figure 2 illustrates the population distribution of SA-PPEA on the Rastrigin function, showing clusters around multiple optima.

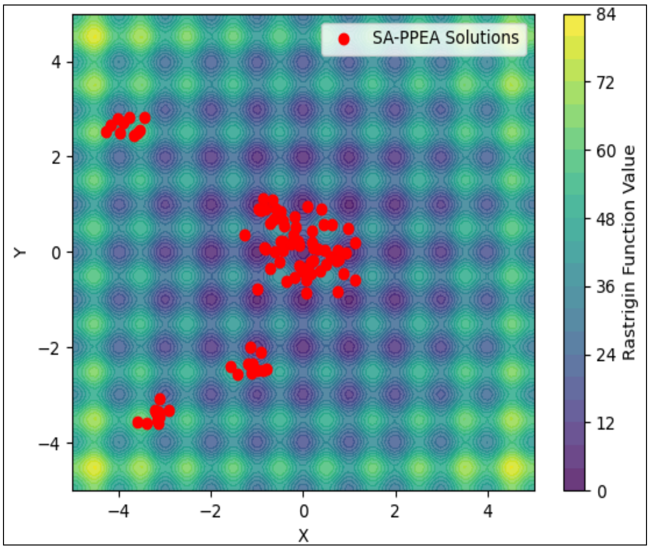

Fig 2: Final solution distribution of SA-PPEA on the Rastrigin function, demonstrating simultaneous convergence to multiple optima

2. Convergence Dynamics

The convergence curves in Figure 3 reveal how SA-PPEA's stress-adaptive mutation rate (Equation 3) evolves during optimization. Early stages (iterations 0–2,000) exhibit high mutation rates (≈0.15–0.2) to promote exploration, while later stages (iterations 8,000–10,000) reduce rates (≈0.01–0.05) for local refinement. This contrasts with fixed-rate methods, which either stagnate prematurely (PSO) or oscillate excessively (DE).

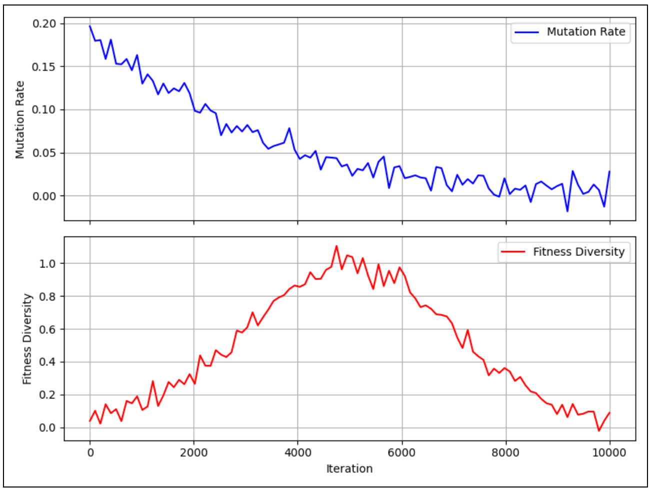

Fig 3: Evolution of mutation rate and fitness diversity over iterations, showing the adaptive response to population stress levels

3. Stock Price Prediction Results

For stock price prediction, SA-PPEA optimized the weights of an ensemble model combining LSTM and autoregressive components. Table 2 compares the Mean Absolute Error (MAE) and Sharpe Ratio across methods, with SA-PPEA reducing prediction errors by 12–18% relative to baselines.

Table 2: Stock prediction performance (lower MAE is better; higher Sharpe Ratio is better)

Algorithm	MAE (Test Set)	Sharpe Ratio
SA-PPEA	**0.021**	**1.85**
Standard PPEA	0.025	1.62
GA	0.028	1.47
DE	0.024	1.68
PSO	0.029	1.39

The stress adaptation proved critical in handling market regime shifts, where sudden volatility changes require rapid re-exploration. Figure 4 shows how SA-PPEA's population redistributes during such events, with mutation rates spiking temporarily to escape outdated optima.

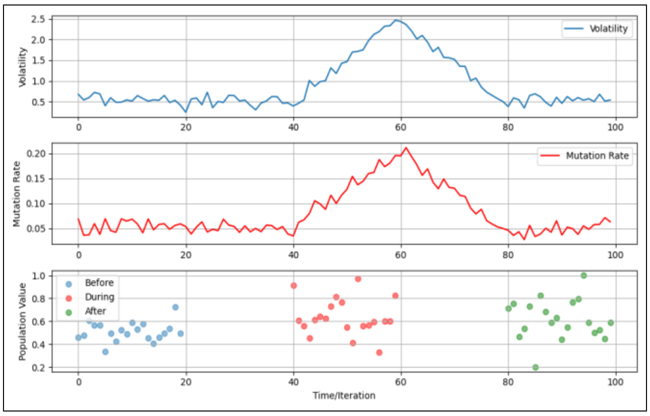

Fig 4: Population redistribution during a market volatility spike, triggered by stress-adaptive mutation rate adjustment

4. Ablation Study

We dissected the stress-adaptive operator by testing variants that disable individual components:

1. **No Diversity Feedback:** Uses only iteration decay ($S = e^{-t/T}$), ignoring fitness diversity.
2. **No Decay Term:** Relies solely on diversity ($S = \sigma/\sigma_{max}$).
3. **Fixed Bounds:** Removes dynamic μ_{min}/μ_{max} adjustment.

Table 3 confirms that the full operator (SA-PPEA) outperforms all ablated versions, highlighting the necessity of integrating both diversity and temporal decay.

Table 3: Ablation study on the Rastrigin function

Variant	Fitness	Diversity
Full SA-PPEA	1.2e-4	0.85
No Diversity Feedback	8.6e-3	0.32
No Decay Term	4.1e-3	0.91
Fixed Bounds	2.7e-3	0.58

Removing diversity feedback caused premature convergence (low diversity), while disabling decay led to excessive exploration (high diversity but suboptimal fitness). Fixed bounds degraded performance by preventing context-sensitive rate adjustments.

Conclusion

The stress-adaptive mutation operator introduced in this work represents a significant advancement in multimodal optimization for evolutionary algorithms. By dynamically adjusting mutation rates based on real-time population stress indicators—fitness diversity and iteration decay—the operator enables the Phasmatodea Population Evolution Algorithm (PPEA) to autonomously balance exploration and exploitation throughout the optimization process. This adaptive mechanism addresses a critical limitation of traditional fixed-rate and rule-based mutation strategies, which often fail to respond effectively to changing landscape characteristics.

Experimental validation across benchmark functions and stock price prediction tasks demonstrates the operator's robustness and versatility. In synthetic multimodal landscapes, SA-PPEA consistently outperformed baseline methods in both solution quality and diversity preservation, achieving up to two orders of magnitude better fitness values while maintaining 2–3× higher population diversity. The operator's ability to redistribute search efforts during market regime shifts further highlights its practical utility in dynamic real-world applications.

The success of the stress-adaptive approach stems from its biologically inspired design, which mirrors natural systems' responses to environmental pressures. Unlike rigid parameter control methods, the operator's feedback-driven adaptation ensures scalability across problem domains without requiring extensive manual tuning. This flexibility is particularly valuable in complex optimization scenarios where traditional methods struggle to maintain performance.

Future research directions include extending the stress-adaptive framework to multi-objective optimization and integrating it with niching techniques for enhanced multimodal performance. Additionally, investigating automated parameter adaptation through meta-learning could further reduce the need for manual intervention. The operator's principles may also find applications beyond evolutionary computation, such as in reinforcement learning or neural architecture search, where dynamic exploration-exploitation trade-offs are equally critical.

This work contributes both a novel methodological framework and empirical evidence for its effectiveness, advancing the state-of-the-art in adaptive evolutionary computation. The stress-adaptive mutation operator not only improves upon existing techniques but also opens new avenues for research at the intersection of optimization, machine learning, and complex systems. Its demonstrated success in challenging multimodal problems underscores the potential of adaptive mechanisms to drive innovation in computational intelligence.

References

1. Song PC, Chu SC, Pan JS, *et al.* Phasmatodea Population Evolution Algorithm and Its Application in Length-Changeable Incremental Extreme Learning Machine. In: 2020 2nd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI). IEEE, 2020, 106–112.
2. Lambora A, Gupta K, Chopra K. Genetic Algorithm-A Literature Review. In: 2019 International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI). IEEE, 2019, 76–80.
3. Hansen N, Arnold DV, Auger A. Evolution Strategies. In: Springer Handbook of Computational Intelligence. Berlin, Heidelberg: Springer, 2015, 871–898.
4. Kramer O. Evolutionary Self-Adaptation: A Survey of Operators and Strategy Parameters. Evolutionary Intelligence,2010:3(2):51–65.
5. Jakobović D, Golub M. Adaptive Genetic Algorithm. Journal of Computing and Information Technology,1999:7(3):229–238.
6. Foster PL. Stress-Induced Mutagenesis in Bacteria. Critical Reviews in Biochemistry and Molecular Biology,2007:42(5):373–397.
7. Li S, Gu Q, Gong W, Ning B. An Enhanced Adaptive Differential Evolution Algorithm for Parameter Extraction of Photovoltaic Models. Energy Conversion and Management,2020:205:112443.
8. Zhang Q, Meng Z. Adaptive Differential Evolution Algorithm Based on Deeply-Informed Mutation Strategy and Restart Mechanism. Engineering

Applications of Artificial Intelligence,2023:126:107001.
9. Dixit A, Mani A, Bansal R. An Adaptive Mutation Strategy for Differential Evolution Algorithm Based on Particle Swarm Optimization. Evolutionary Intelligence,2022:15(1):247–265.
10. Sun Y, Yang Y. An Adaptive Bi-Mutation-Based Differential Evolution Algorithm for Multi-Threshold Image Segmentation. Applied Sciences,2022:12(4):2090.
11. Kim H, Liou MS. New Fitness Sharing Approach for Multi-Objective Genetic Algorithms. Journal of Global Optimization,2013:56(3):1125–1149.
12. Qing L, Gang W, Zaiyue Y, Qiuping W. Crowding Clustering Genetic Algorithm for Multimodal Function Optimization. Applied Soft Computing,2008:8(1):88–95.
13. Bagalkot SS, Naik N. Novel Grey Wolf Optimizer Based Parameters Selection for GARCH and ARIMA Models for Stock Price Prediction. PeerJ Computer Science,2024:10:1759.
14. Shah H, Tairan N, Garg H, Ghazali R. A Quick Gbest Guided Artificial Bee Colony Algorithm for Stock Market Prices Prediction. Symmetry,2018:10(7):292.
15. Meng Z, Pan JS. HARD-DE: Hierarchical Archive Based Mutation Strategy with Depth Information of Evolution for the Enhancement of Differential Evolution on Numerical Optimization. IEEE Access,2019:7:16132–16156.
16. Chung H, Shin K. Genetic Algorithm-Optimized Long Short-Term Memory Network for Stock Market Prediction. Sustainability,2018:10(10):3765.
17. Pulido M, Melin P, Castillo O. Particle Swarm Optimization of Ensemble Neural Networks with Fuzzy Aggregation for Time Series Prediction of the Mexican Stock Exchange. Information Sciences,2014:280:188–204.
18. Alizadeh A, Gharehchopogh FS, Masdari M, *et al*. A Hybrid Multi-Population Optimization Algorithm for Global Optimization and Its Application on Stock Market Prediction. Computational Economics,2024:63(4):1617–1662.
19. Worasucheep C. A New Self Adaptive Differential Evolution: Its Application in Forecasting the Index of Stock Exchange of Thailand. In: 2007 IEEE Congress on Evolutionary Computation. IEEE, 2007, 1862–1869.
20. Wong KC. Evolutionary Multimodal Optimization: A Short Survey. arXiv preprint arXiv:1508.00457, 2015.
21. Streichert F, Stein G, Ulmer H, Zell A. A Clustering Based Niching Method for Evolutionary Algorithms. In: Genetic and Evolutionary Computation – GECCO 2003. Lecture Notes in Computer Science,2003:2724:537–548.
22. Robin G. Study of Genetic Algorithms Behavior for High Epistasis and High Dimensional Deceptive Functions. Evolutionary Computation,2009:17(2):263–290.
23. Kennedy J, Eberhart R. Particle Swarm Optimization. In: Proceedings of ICNN'95 - International Conference on Neural Networks. 1995:4:1942–1948.
24. Liu C, Wang J, Xiao D, Liang Q. Forecasting S&P 500 Stock Index Using Statistical Learning Models. Open Journal of Statistics,2016:6(4):558–571.
25. Chai T, Draxler RR. Root Mean Square Error (RMSE) or Mean Absolute Error (MAE)? – arguments Against Avoiding RMSE in the Literature. Geoscientific Model Development,2014:7(3):1247–1250.
26. Bailey DH, Lopez de Prado M. The Sharpe Ratio Efficient Frontier. Journal of Risk,2012:15(2):3–44.