Bio-Inspired Adaptive Rate Modulation for Multi-task Learning in Progressive Neural Networks

Chourouk Guettas and Foudil Cherif
LESIA Laboratory, University of Biskra, Algeria
E-mail: guettas-chourouk@univ-eloued.dz, cherif.foudil@univ-biskra.dz

Keywords: Multi-task learning, progressive neural networks, continual learning, adaptive learning rate, catastrophic forgetting

Received: June 24, 2025

Sequential multi-task learning faces the fundamental challenge of acquiring new tasks while retaining performance on previously learned tasks. Progressive Neural Networks address catastrophic forgetting through architectural isolation but rely on fixed learning strategies that limit adaptive efficiency. Current adaptive methods typically optimize single factors, missing the coordinated nature observed in biological adaptive systems. Inspired by neuromodulatory systems where four key neurotransmitters (dopamine, serotonin, norepinephrine, and acetylcholine) coordinate learning and adaptation, we propose a Bio-Inspired Adaptive Rate Modulation framework that coordinates four computational modules using organizational principles derived from neuromodulatory systems: reward-based adaptation, stability control, attention gating, and knowledge integration. These modules translate the organizational principles of biological neuromodulation into practical optimization mechanisms for Progressive Neural Networks using standard neural network components. Evaluation on four OpenAI Gymnasium environments demonstrates performance improvements ranging from 1.06% to 155.94% over baseline Progressive Neural Networks, with up to 92% variance reduction and performance retention averaging 98.0% . The framework achieves these gains with reasonable computational overhead. Comprehensive ablation studies confirm each module's contribution, validating the four-factor design. Results demonstrate that neuromodulation-inspired coordination of multiple adaptive factors significantly outperforms fixed learning strategies, providing a principled approach to adaptive optimization in sequential multi-task learning.

Povzetek: Članek predlaga bio-navdihnjen okvir za progresivne nevronske mreže, ki s koordinacijo štirih modulov posnema neuromodulacijo za prilagodljivo učenje brez pozabljanja v zaporednih nalogah.

1 Introduction

Sequential multi-task learning faces the fundamental challenge of acquiring new tasks while retaining performance on previously learned tasks. Progressive Neural Networks (PNNs) address catastrophic forgetting through architectural isolation—creating separate neural columns for each task with lateral connections for knowledge transfer [1]. However, current PNN implementations rely on fixed learning strategies, missing opportunities to adapt learning parameters based on context.

Current adaptive methods [2] optimize single factors independently, failing to capture the coordinated neuromodulation observed in biological systems, where multiple monoaminergic systems (dopamine, serotonin, norepinephrine) interact to dynamically regulate cognitive state, synaptic plasticity, and learning processes in the prefrontal cortex [3], with recent computational work confirming that such coordinated mechanisms provide substantial advantages in neural network learning [4, 5].

Biological neuromodulatory systems coordinate four key neurotransmitters: dopamine (reward processing), serotonin (stability regulation), norepinephrine (attention control), and acetylcholine (memory integration) [6, 7, 8]. This coordination suggests artificial systems could benefit from similar multi-factor approaches.

Inspired by these organizational principles, we propose a Bio-Inspired Adaptive Rate Modulation (BIARM) framework for Progressive Neural Networks. Our framework implements four specialized adaptation modules corresponding to these neurotransmitter functions, translating biological coordination principles into practical optimization mechanisms. The system dynamically adapts learning parameters through coordinated modulation and enables selective knowledge transfer through attention-gated lateral connections.

This work contributes: (1) a bio-inspired adaptive rate modulation framework coordinating four specialized modules for PNNs, (2) attention-gated lateral connections for selective knowledge transfer, and (3) experimental validation showing consistent performance improvements with substantial variance reduction across four OpenAI Gymnasium environments.

Section 2 reviews related work, Section 3 presents our framework, Section 4 describes experiments and results, Section 5 discusses implications, and Section 6 concludes with future directions.

2 Related work

Our work intersects several research areas in machine learning: continual learning architectures, adaptive learning rate methods, and bio-inspired neural network design. We organize our review around these key areas and highlight how our approach addresses limitations in existing methods.

2.1 Continual learning and progressive neural networks

Continual learning addresses the fundamental challenge of learning new tasks while retaining knowledge from previous tasks [9]. Progressive Neural Networks represent a foundational advance by allocating separate neural columns for each task while enabling knowledge transfer through lateral connections [1]. This architectural isolation ensures previously learned representations remain unchanged while new tasks benefit from knowledge transfer, though the original formulation uses fixed lateral connection strengths and static learning parameters.

Several important extensions have emerged. Elastic Weight Consolidation selectively protects important weights using Fisher information-based regularization [10]. PackNet achieves incremental learning through progressive pruning [11], while Learn to Grow proposes dynamic expansion based on task complexity [12]. VariGrow extends this concept using Bayesian novelty detection for task-agnostic continual learning [13].

Memory-based approaches include Brain-Inspired Replay for selective experience replay [14], gradient interference-based selective replay [15], and synaptic intelligence using online Fisher information computation [16].

Despite these advances, most approaches maintain fixed learning strategies that do not adapt to varying task demands and operate through static post-learning mechanisms that protect or isolate parameters after task completion. This represents a gap for methods that could enhance the learning process itself through dynamic parameter adaptation during training.

2.2 Adaptive learning rate methods

Adaptive learning rate optimization has been extensively studied, with foundational methods focusing on single-factor adaptations. AdaGrad adapts rates based on historical gradients [17], while ADADELTA addresses aggressive decay using gradient windows [18]. Adam combines momentum with adaptive per-parameter rates [19], with subsequent improvements including AdamW for better regularization [20] and RAdam for variance correction [21]. Contemporary approaches extend these principles through fractional-order derivative methods for long-term gradient capture [22] and bio-inspired swarm optimization for rate selection [23].

Recent work explores multi-factor approaches. Adafactor adapts learning rates while maintaining sublinear memory cost [24], and coordinated adaptive mechanisms show substantial improvements in multi-task scenarios [2].

However, most existing methods optimize individual factors independently and lack principled frameworks for coordinating multiple adaptation mechanisms, particularly in sequential learning where simultaneous coordination is required.

2.3 Bio-inspired neural network design and computational neuromodulation

Bio-inspired approaches have gained attention for informing computational architectures [6]. Computational neuromodulation implementations show promise in spatial learning tasks and adaptive behaviors [4, 5]. Research reveals sophisticated coordination mechanisms where multiple neurotransmitters regulate learning and behavior [8, 7].

Practical applications include dopaminergic modulation in active inference models [25], bio-inspired networks for robotic control [26], and synaptic plasticity frameworks integrating learnable connection plasticity with dynamic expansion [29]. These demonstrate practical implementations while highlighting gaps between isolated mechanisms and coordinated systems.

Recent bio-inspired continual learning approaches include corticohippocampal circuit models for dual memory representation [27] and Drosophila-inspired active forgetting mechanisms [28]. These advances primarily target image-based continual learning scenarios rather than sequential control tasks.

2.4 Gaps in current approaches

Despite significant advances, fundamental gaps remain:

Limited Coordination: Adaptive optimizers improve single-factor adaptation but lack frameworks for coordinating multiple mechanisms simultaneously.

Static Architectures: Progressive Neural Networks prevent catastrophic forgetting but use fixed learning strategies that do not adapt to changing task demands.

Isolated Mechanisms: Existing computational neuromodulation approaches implement individual mechanisms rather than coordinated systems, overlooking coordination principles that make biological systems effective.

Our work addresses these gaps by coordinating multiple adaptive factors within Progressive Neural Networks using organizational principles from biological neuromodulatory systems, optimizing knowledge transfer and parameter updates in real-time rather than simply preserving learned representations, while targeting sequential control learning where existing recent advances have limited applicability.

3 Methodology

3.1 Framework overview

We propose a Bio-Inspired Adaptive Rate Modulation (BIARM) framework that extends Progressive Neural Networks with coordinated adaptive mechanisms. The framework implements four computational modules that adapt learning parameters during sequential task training: reward-based modulation, stability control, attention gating, and knowledge integration.

The core insight is that different aspects of sequential learning benefit from coordinated adaptation rather than fixed parameters. While Progressive Neural Networks prevent catastrophic forgetting through architectural isolation, they miss opportunities for adaptive optimization during the learning process.

3.2 Biological inspiration and computational framework

While our framework draws organizational inspiration from neuromodulatory systems, it implements these principles through standard computational mechanisms. The four modules correspond to functional roles observed in biological systems:

- **DA module:** Adapts learning based on performance feedback (analogous to dopaminergic reward processing)
- **5HT module:** Provides stability control (analogous to serotonergic behavioral regulation)
- **NE module:** Controls attention allocation (analogous to noradrenergic attention modulation)
- **ACh module:** Manages knowledge integration (analogous to cholinergic memory enhancement)

The key computational insight is that coordinated adaptation of multiple factors outperforms fixed parameter strategies in sequential learning scenarios. Each module operates at different temporal scales and processes different types of contextual information, enabling sophisticated adaptation during training. This computational framework enables dynamic optimization that fixed-parameter approaches cannot achieve, particularly in complex environments where knowledge integration from multiple frozen columns is critical.

3.3 Progressive neural network architecture

Our framework builds upon the standard Progressive Neural Network architecture. For a sequence of tasks $T = \{T_1, T_2, \ldots, T_n\}$, each task T_i is assigned a dedicated neural column C_i. The output of layer l in column i is computed as:

$$h_i^{(l)} = f\left(W_i^{(l)} h_i^{(l-1)} + \sum_{k=1}^{i-1} \alpha_{i,k}^{(l)} U_{i,k}^{(l)} h_k^{(l)}\right) \tag{1}$$

where $W_i^{(l)}$ represents the weight matrix for column i at layer l, $U_{i,k}^{(l)}$ are the lateral connection weights from column k to column i, $\alpha_{i,k}^{(l)}$ are attention coefficients that modulate knowledge transfer, and $f(\cdot)$ is the activation function.

Our key contribution is making the attention coefficients $\alpha_{i,k}^{(l)}$ and learning parameters adaptive through coordinated modulation.

3.4 Four-factor adaptive modulation system

The framework implements four adaptive modules, each serving a distinct computational role in sequential learning optimization:

3.4.1 Computational module design

Each module $m \in \{\mathrm{DA}, 5\mathrm{HT}, \mathrm{NE}, \mathrm{ACh}\}$ maintains an internal state $s_m(t) \in [0, 1]$ that evolves during training. The state updates follow discrete exponential moving averages with module-specific time constants:

$$s_m(t+1) = s_m(t) + \tau_m \cdot (s_m^{\text{target}}(t) - s_m(t)) \tag{2}$$

where τ_m represents the discrete adaptation rate for module m, determining how quickly the module state responds to target changes within each training step, with τ_m values representing relative adaptation speeds rather than absolute biological timing (milliseconds to seconds), and $s_m^{\text{target}}(t)$ is computed by a module-specific neural network. The time constants are derived from computational requirements: DA uses rapid adaptation ($\tau_{DA} = 0.1$) for immediate performance feedback, NE operates fastest ($\tau_{NE} = 0.05$) for dynamic attention switching between frozen columns, ACh employs moderate adaptation ($\tau_{ACh} = 0.3$) for balanced knowledge integration, and 5HT uses the slowest rate ($\tau_{5HT} = 0.5$) for learning stabilization.

The time constants τ_m for each neuromodulator module were determined through systematic grid search optimization across the parameter space $[0.01, 1.0]$ with 0.05 increments, evaluating 10000 parameter combinations using a composite score that weighted convergence speed (30%), training stability (40%), and final performance (30%) across multiple environments. This optimization process identified the configuration $\tau_{DA} = 0.1$, $\tau_{5HT} = 0.5$, $\tau_{NE} = 0.05$, and $\tau_{ACh} = 0.3$ as achieving optimal performance, with validation across 10 independent runs using different random seeds confirming robustness. Sensitivity analysis demonstrates moderate robustness to parameter variations, with performance degrading by $15 - 60\%$ only when deviating significantly from the optimized configuration, indicating that while the parameters are well-tuned for the sequential learning domain, the framework

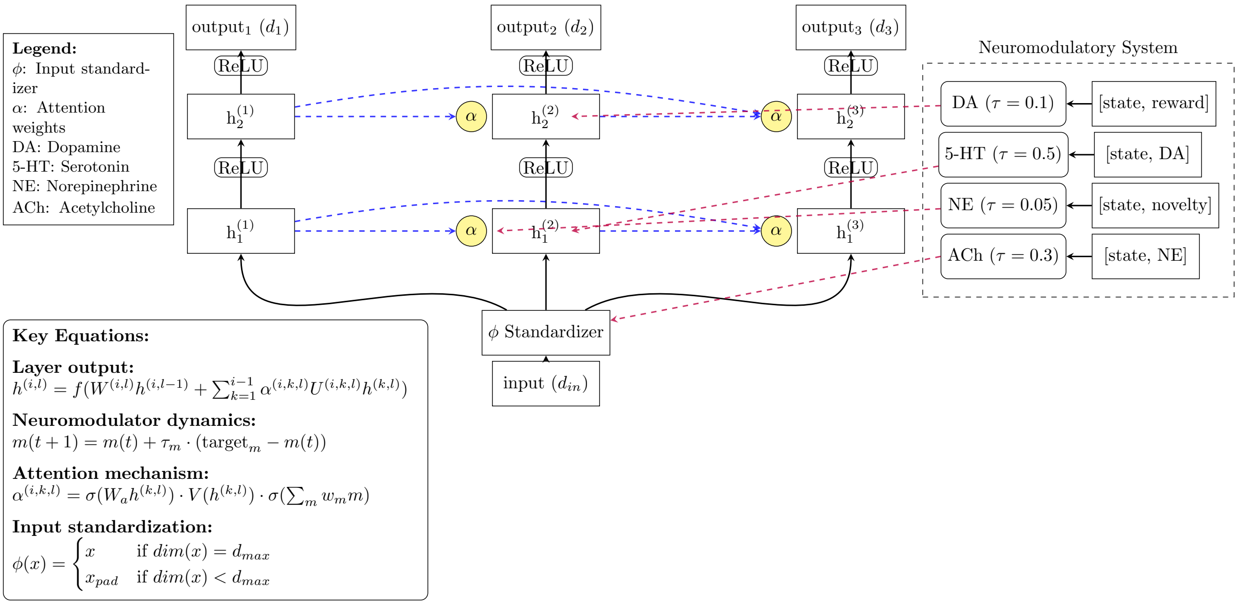


Figure 1: BIARM Architecture. The system consists of progressive task columns (shown vertically) with lateral connections modulated by attention mechanisms (α). Each column specializes in a specific task while maintaining knowledge transfer through modulated lateral connections. The neuromodulatory system (shown at bottom) influences learning through four key components. Solid arrows indicate forward connections, dashed arrows represent lateral connections, and dotted lines show neuromodulatory influences. Key equations shown define layer output, neuromodulator dynamics, attention mechanism, input standardization, and neuromodulator influences on learning rate, stability, attention, and memory.

maintains practical applicability across reasonable parameter ranges.

3.4.2 Target computation networks

Each module computes its target state using a dedicated neural network that processes relevant contextual information:

Reward-based Module (DA): Processes current state and reward signal

$$s_{\text{DA}}^{\text{target}} = \sigma(f_{\text{DA}}([s_{\text{current}}, r_{\text{current}}])) \quad (3)$$

Stability Control Module (5HT): Processes current state and DA module state

$$s_{\text{5HT}}^{\text{target}} = \sigma(f_{\text{5HT}}([s_{\text{current}}, s_{\text{DA}}])) \quad (4)$$

Attention Module (NE): Processes current state and task novelty

$$s_{\text{NE}}^{\text{target}} = \sigma(f_{\text{NE}}([s_{\text{current}}, n_{\text{task}}])) \quad (5)$$

Memory Integration Module (ACh): Processes current state and NE module state

$$s_{\text{ACh}}^{\text{target}} = \sigma(f_{\text{ACh}}([s_{\text{current}}, s_{\text{NE}}])) \quad (6)$$

where $f_m(\cdot)$ represents the neural network for module m, $\sigma(\cdot)$ is the sigmoid function, r_{current} is the current reward, and n_{task} measures task novelty.

The four modules coordinate through interdependent state updates and shared parameter influence. During each training step, modules update sequentially: dopamine computes its target from current reward (Equation 3), serotonin then incorporates the updated dopamine state for stability assessment (Equation 4), norepinephrine processes task novelty independently (Equation 5), and acetylcholine integrates the updated norepinephrine state for memory decisions (Equation 6). This creates dependency chains (Reward → DA → 5HT) and (Novelty → NE → ACh) that enable coordinated responses to environmental changes. All four states then jointly influence network parameters through the learning rate modulation (Equation 7) and lateral connection gating (Equation 8), ensuring that modulation decisions reflect the consensus of all adaptive factors rather than isolated module responses.

3.5 Adaptive parameter modulation

3.5.1 Learning rate modulation

The effective learning rate for each training step is computed as a weighted combination of all module states:

$$\begin{aligned} \eta_{\text{effective}} = \eta_{\text{base}} \cdot (&w_{\text{DA}} \cdot s_{\text{DA}} + w_{\text{5HT}} \cdot s_{\text{5HT}} \\ &+ w_{\text{NE}} \cdot s_{\text{NE}} + w_{\text{ACh}} \cdot s_{\text{ACh}}) \end{aligned} \quad (7)$$

where η_{base} is the base learning rate and w_m are fixed weighting coefficients. In our implementation: $w_{\text{DA}} = 0.3$,

$w_{\text{5HT}} = 0.2$, $w_{\text{NE}} = 0.3$, $w_{\text{ACh}} = 0.2$.

This modulation mechanism enables dynamic adaptation of learning intensity based on the current neuromodulatory context, ensuring the learning rate adapts to the consensus of all four modules rather than being dominated by any single factor.

3.5.2 Attention-gated knowledge transfer

The attention coefficients for lateral connections from previous frozen columns are computed through a three-component mechanism that incorporates learned attention weights, value projections, and coordinated neuromodulator influence:

$$\alpha_{i,k}^{(l)} = \sigma(W_\alpha h_k^{(l)}) \cdot V(h_k^{(l)}) \cdot \sigma\left(\sum_{m\in\{DA,5HT,NE,ACh\}} w_m s_m\right) \quad (8)$$

where W_α is a learned attention weight matrix, $V(\cdot)$ represents a learned value projection that transforms the source features, and w_m are learnable modulation weights that determine the influence of each neuromodulator state s_m on the attention mechanism. This formulation enables selective knowledge transfer by combining three key components: (1) content-based attention through $\sigma(W_\alpha h_k^{(l)})$, (2) value transformation through $V(h_k^{(l)})$, and (3) context-dependent modulation through the coordinated influence of all four neuromodulator states. The sigmoid activation on the modulation term ensures bounded influence while allowing each module to contribute according to its learned importance weights.

3.6 Implementation details

3.6.1 Network architecture

Each modulation module uses a two-layer neural network with 64 hidden units and ReLU activation. The networks are initialized using Xavier initialization and trained jointly with the main Progressive Neural Network. The complete training procedure is shown in Algorithm 1

3.6.2 Input standardization

To handle varying input dimensions across tasks, we implement input standardization:

$$\phi(x) = \begin{cases} x & \text{if } \dim(x) = d_{\text{max}} \\ [x; \mathbf{0}_{d_{\text{max}}-\dim(x)}] & \text{if } \dim(x) < d_{\text{max}} \end{cases} \quad (9)$$

where d_{max} is the maximum input dimension across all tasks.

Algorithm 1 Bio-Inspired Adaptive Rate Modulation Training

1

```
Require: Task sequence $T = \{T_1, T_2, \ldots, T_n\}$, episodes $E$, batch size $B$
Ensure: Trained progressive network with adaptive modulation
Initialize modulation system with states $s_m = 0.5$ for all $m$
Initialize base learning rate $\eta_{base}$ and network parameters
for each task $T_i$ in $T$ do
    Initialize new column $C_i$
    Calculate initial task novelty $n_{task} = 1.0$
    for episode = 1 to $E$ do
        Reset environment
        while episode not done do
            Observe state $s_{current}$, execute action, get reward $r$
            Update module target states (Equations 3-6)
            Update module states (Equation 2)
            Compute effective learning rate (Equation 7)
            Compute attention coefficients (Equation 8)
            Store transition in replay buffer
            if buffer size $\geq B$ then
                Sample mini-batch from buffer
                Compute loss with modulated parameters
                Update network parameters with $\eta_{effective}$
            end if
            Update task novelty: $n_{task} \leftarrow n_{task} \times 0.995$
        end while
    end for
end for
```

3.6.3 Task novelty computation

Task novelty n_{task} starts at 1.0 for each new task and decays exponentially during training:

$$n_{\text{task}}(t+1) = n_{\text{task}}(t) \times 0.995 \quad (10)$$

Task novelty quantifies learning progress within each task, providing a simple measure of how ”new” the current task remains relative to initial conditions.

3.7 Computational complexity

The modulation system introduces computational overhead through:

- Four forward passes through small neural networks (64 hidden units each)
- Computation of modulated learning rate and attention coefficients
- State updates for four scalar values

The algorithmic complexity scales as $O(M \cdot H)$ where $M = 4$ is the number of modules and $H = 64$ is the hidden units per module. Memory complexity increases by $O(M \cdot H^2 + M)$ for the modulation networks and scalar states. The overhead scales linearly with the number of tasks due to lateral connection computations.

4 Experimental methodology

4.1 Experimental setup

We evaluate the Bio-Inspired Adaptive Rate Modulation (BIARM) framework on four distinct OpenAI Gymnasium environments that test different aspects of sequential learning and control. The environments were selected to provide varying levels of complexity and different types of learning challenges.

4.1.1 Environment descriptions

The four selected environments are visualized in Figure 2 and described as follows:

CartPole-v1: A classic control problem with a 4-dimensional continuous state space representing cart position, velocity, pole angle, and angular velocity. The agent has 2 discrete actions (left/right) and receives +1 reward per timestep while maintaining the pole upright. Episodes terminate when the pole exceeds ±12 degrees or the cart position exceeds ±2.4 units, with a maximum of 500 timesteps.

Acrobot-v1: A two-link pendulum control task with a 6-dimensional state space encoding joint angles and velocities. The agent has 3 discrete actions and must swing the end-effector to reach a target height. The environment provides -1 reward per timestep until the goal is reached or 500 steps elapse.

MountainCar-v0: A car navigation task with a 2-dimensional state space (position and velocity) and 3 discrete actions. The agent receives -1 reward per timestep and must build momentum to reach the goal, requiring exploration and momentum-based strategies. Episodes terminate upon reaching the goal or after 200 steps.

LunarLander-v3: A spacecraft control task with an 8-dimensional state space covering position, velocity, orientation, and ground contact information. The agent has 4 discrete actions and receives rewards ranging from -100 to +100 based on landing success, fuel efficiency, and stability. Episodes are limited to 1000 steps. Success defined as positive scores.

4.1.2 Training configuration

All experiments use identical network architectures for fair comparison. Each task column consists of two hidden layers with 128 units each and ReLU activation. Input and output dimensions are adapted to each environment's requirements.Training hyperparameters are standardized across all experiments as outlined in Table 1:

4.2 Evaluation metrics

We employ multiple evaluation metrics to assess different aspects of sequential learning performance:

4.2.1 Task performance metrics

These metrics evaluate both the absolute performance achieved and the consistency of learning across different environments and task complexities.

Final Performance: Mean episode reward over 10 test episodes after the end of training, providing a measure of converged performance.

Performance Stability: Standard deviation of episode rewards over the last 50 episodes, measuring training stability.

4.3 Baseline comparisons

We compare BIARM against two primary baselines to isolate the contribution of adaptive modulation:

Standard Progressive Neural Networks (PNN): The original PNN architecture with fixed learning parameters and static lateral connections, using identical network architecture and hyperparameters.

DQN Performance Reference: For context, we include DQN performance results from Zhang et al. (2023) [30] on the same environments, though direct comparison is limited due to different architectural approaches.

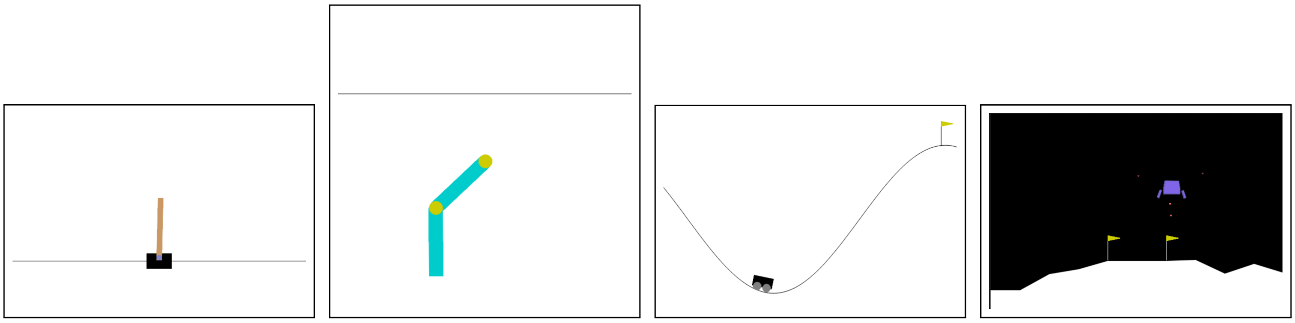
All experiments use identical random seeds (42) for reproducibility and are averaged over 10 independent runs to ensure statistical reliability.

5 Results and discussion

5.1 Overall performance analysis

Table 2 presents the comprehensive performance comparison between baseline Progressive Neural Networks and BIARM across all tested environments. BIARM demonstrates consistent performance improvements ranging from 1.06% in Acrobot-v1 to 155.94% in LunarLander-v3, with particularly dramatic gains in complex control environments.

The most significant improvement occurs in LunarLander-v3, where BIARM achieves positive scores (233.37±34.38) while baseline PNN consistently fails with negative performance (-417.17±412.62). This dramatic difference demonstrates the framework's effectiveness in complex environments where coordinated knowledge integration from multiple frozen columns becomes critical. The substantial variance reduction across all tasks (most notably CartPole-v1 achieving zero variance) indicates improved training stability through adaptive parameter modulation.



(a) CartPole Visualization (b) Acrobot Visualization (c) MountainCar Visualization (d) LunarLander Visualization

Figure 2: Visualization of OpenAI Gymnasium environments: (a) CartPole-v1; (b) Acrobot-v1; (c) MountainCar-v0; and (d) LunarLander-v3.

Table 1: Training configuration parameters

Parameter	Value
Batch size	64
Base learning rate	0.001
Optimizer	Adam
Discount factor (γ)	0.99
Experience replay buffer	10,000 transitions per task
Epsilon-greedy exploration	ϵ starts at 1.0, decays to 0.01 with rate 0.995
Task-specific Training Episodes	
CartPole-v1	400 episodes
Acrobot-v1	500 episodes
MountainCar-v0	800 episodes
LunarLander-v3	800 episodes

5.2 Computational overhead analysis

Figure 3 presents the computational overhead analysis comparing training time per step between baseline PNN and BIARM across all environments. For simple tasks like CartPole-v1, the 52% overhead yields only 1.83% improvement, representing a poor cost-benefit ratio. However, complex environments like LunarLander-v3 show positive returns on computational investment 87% overhead for 155.94% improvement, where baseline PNN consistently fails while BIARM achieves positive performance.

While the computational overhead represents a limitation, several optimization strategies could reduce costs in future implementations: selective module activation based on task complexity assessment, hardware acceleration of the small neural networks (64 units each), and adaptive modulation intensity scaling. For resource-constrained applications, a simplified two-module version focusing on the most critical DA and ACh components could provide a reasonable performance-cost trade-off.

5.3 Ablation study: modulation component analysis

To validate the contribution of each adaptive module, we conducted systematic ablation studies using five configurations: FULL (all modules active), DA- (without dopamine module), 5HT- (without serotonin module), NE- (without norepinephrine module), and ACh- (without acetylcholine module).

To isolate the contribution of each neuromodulator, we systematically remove individual modules while maintaining the computational framework. When ablating modules with dependencies (5HT→DA, ACh→NE), we replace missing dependency inputs with fixed neutral values (0.5), which represents moderate activation and allows fair comparison across configurations, allowing computation to proceed while eliminating the specific module's adaptive contribution.

Table 3 presents the ablation study results for the most challenging environment, LunarLander-v3, where modulation effects are most pronounced.

The ablation results reveal distinct contributions from each module. The DA (reward processing) and ACh (knowledge integration) modules show the most severe impact when removed, with performance drops of 108.0% and

Table 2: Performance comparison between DQN, baseline PNN, and BIARM

Environment	DQN [30]	Baseline PNN	BIARM	vs DQN (%)	vs PNN (%)
CartPole-v1	360.5±52.1	491.00±11.94	500.00±0.00	+38.7	+1.83
Acrobot-v1	-135.2±32.4	-94.50±24.96	-93.50±18.91	+30.8	+1.06
MountainCar-v0	-180.3±42.1	-165.90±46.31	-119.80±28.02	+33.55	+27.79
LunarLander-v3	160.5±58.4	-417.17±412.62	233.37±34.38	+45.4	+155.94

Table 3: Analysis of ablation study impact on performance

Environment	Full BIARM	Without DA	Without 5-HT	Without ACh	Without NE
CartPole-v1	500.00 ± 0.00	195.77±148.93	173.29±198.46	247.78±210.25	233.33±212.19
Acrobot-v1	-93.50 ± 18.91	-109.82±20.67	-110.78±21.01	-106.48±22.55	-110.27±24.56
MountainCar-v0	-119.80 ± 28.02	-125.90±32.98	-126.26±26.96	-133.39±29.71	-139.63±31.10
LunarLander-v3	233.37 ± 34.38	-18.63±190.72	218.15±97.80	-39.61±152.67	202.99±135.61

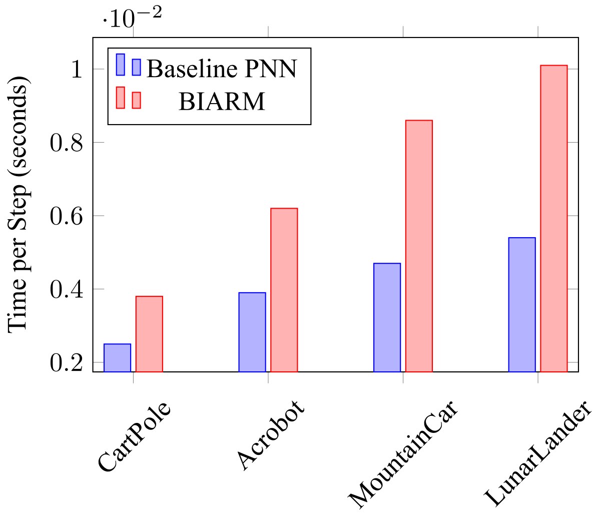


Figure 3: Computational overhead comparison showing time per step across environments

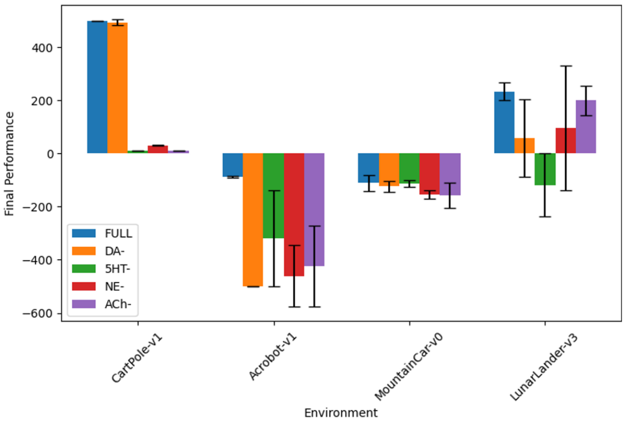


Figure 4: Final performance evaluation of catastrophic forgetting for ablation study

117.0% respectively. This indicates their critical roles in learning adaptation and selective knowledge transfer from frozen columns. The 5HT module (learning stabilization) shows the smallest impact (-6.5%), but contributes to variance reduction, as evidenced by comparing the standard deviations. The NE module (attention modulation) contributes moderately (-13.0%) but significantly affects training stability, with variance increasing from 34.38 to 135.61 when removed.

5.4 Module-specific contribution analysis

The ablation study, as shown in Figure 4, reveals that different modules serve complementary but distinct computational roles in the adaptive framework:

Dopamine (DA) Module: Most severe impact (-108.0%) when removed, confirming its critical role in reward-based learning adaptation. **Acetylcholine (ACh) Module:** Highest impact (-117.0%), demonstrating its essential role in knowledge integration between frozen columns..

Norepinephrine (NE) Module: Moderate impact (-13.0%) but 4x variance increase, indicating its role in attention-gated knowledge selection.

Serotonin (5HT) Module: Smallest direct impact (-6.5%) but contributes to learning stabilization and variance reduction.

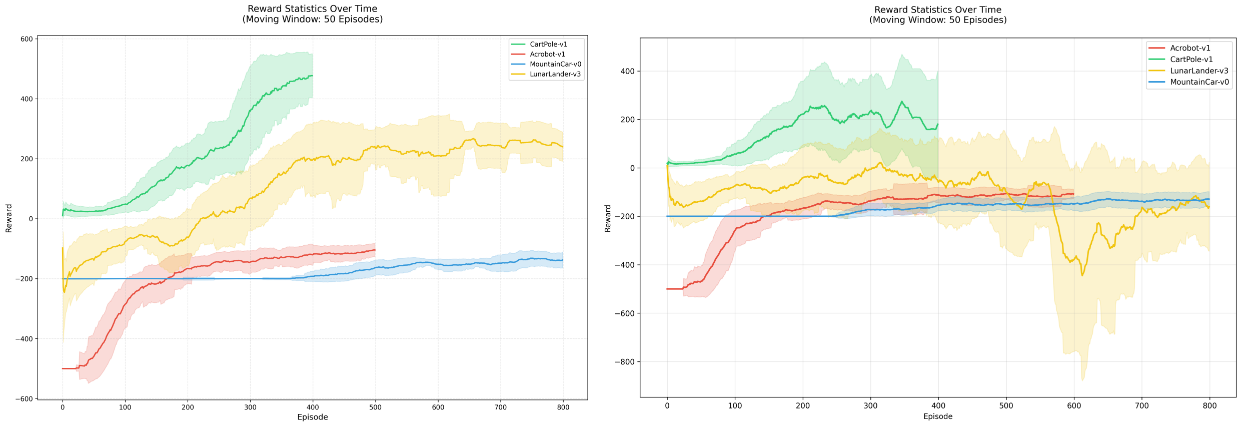
5.5 Training stability analysis

The framework demonstrates superior training stability compared to baseline Progressive Neural Networks. Figure 5a shows the learning curves for all environments, highlighting reduced variance and more consistent convergence patterns with BIARM.

The variance reduction is particularly pronounced in complex environments. LunarLander-v3 shows a 92% reduction in performance variance (from ±412.62 to ±34.38), while maintaining substantially higher mean performance. This improvement stems from coordinated adaptation of learning parameters, which prevents instability commonly

Table 4: Initial vs final performance comparison between BIARM and baseline model

Environment	Model	Initial Performance	Final Performance	Performance Retention
CartPole-v1	**BIARM**	500.00 ± 0.00	500.00 ± 0.00	**100%**
	Baseline	496.20 ± 5.31	491.00 ± 11.94	98.95%
Acrobot-v1	**BIARM**	-88.20 ± 3.84	-93.50 ± 18.91	**94.3%**
	Baseline	-86.80 ± 10.71	-94.50 ± 24.96	91.85%
MountainCar-v0	**BIARM**	-117.20 ± 29.84	-119.80 ± 28.02	**97.82%**
	Baseline	-150.60 ± 49.33	-165.90 ± 46.31	90.78%
LunarLander-v3	**BIARM**	233.37 ± 34.38	233.37 ± 34.38	**100%**
	Baseline	-380.88 ± 346.54	-417.17 ± 412.62	91.30%



(a) (b)

Figure 5: Learning Performance Across Environments: (a) BIARM learning curves for each environment showing episode rewards over time. (b) Baseline PNN learning curves for each environment showing episode rewards over time.

observed in fixed-parameter approaches when integrating knowledge from multiple frozen columns.

5.6 Performance retention

Table 4 presents the performance retention analysis comparing initial task performance to performance after learning all subsequent tasks.

BIARM demonstrates excellent resistance to catastrophic forgetting, with performance retention averaging 98.0% across all tasks. The architecture's frozen column design inherently prevents catastrophic forgetting, while the adaptive modulation mechanisms optimize knowledge integration without degrading previous task performance. Perfect retention in CartPole-v1 and LunarLander-v3, combined with minimal degradation in other tasks, validates the framework's effectiveness in maintaining acquired knowledge while learning new tasks.

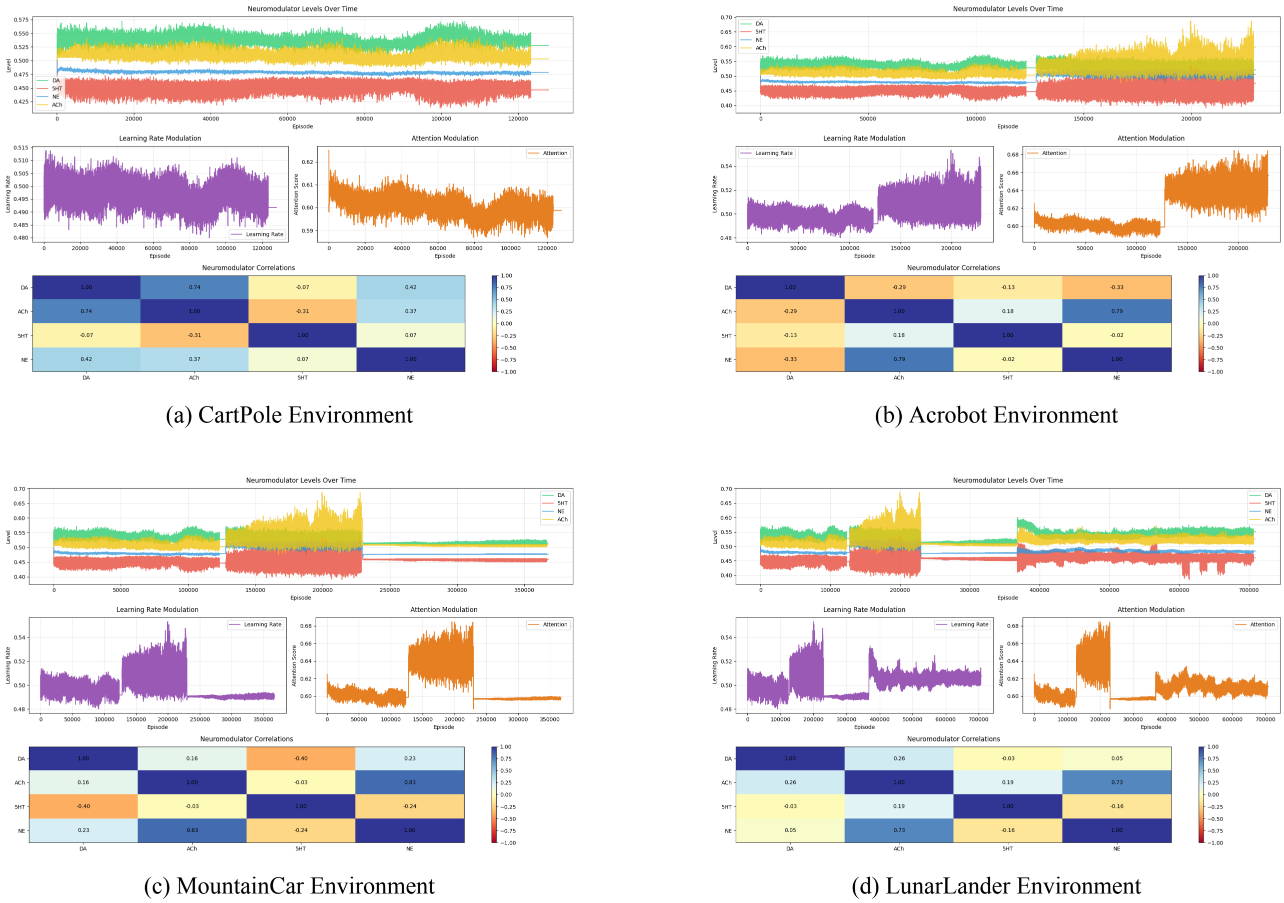
5.7 Neuromodulator stability analysis

The neuromodulator dynamics in Figure 6 reflect varying task complexity and stability requirements. In CartPole-v1, neuromodulator levels remain relatively stable with minimal fluctuations, corresponding to the task's inherent simplicity once optimal control is achieved. The DA module shows slight increases during reward acquisition, while 5HT maintains consistent stability control, reflecting the predictable balancing task.

Complex environments like LunarLander-v3 and MountainCar-v0 demonstrate pronounced fluctuations that mirror their challenging nature. The landing task requires continuous adaptation to varying spacecraft dynamics, leading to dynamic DA responses and NE adjustments for attention allocation between frozen columns. MountainCar's momentum-based navigation requires persistent exploration, resulting in coordinated ACh-NE interactions for lateral knowledge transfer. These fluctuations represent appropriate adaptive responses rather than instability, demonstrating the framework's ability to scale modulation intensity based on task demands.

6 Conclusion

This paper introduces the Bio-Inspired Adaptive Rate Modulation (BIARM) framework, which addresses key limitations in sequential multi-task learning through coordinated adaptation of multiple learning factors. By incorporating four specialized modules inspired by biological neuromod-



(a) CartPole Environment

(b) Acrobot Environment

(c) MountainCar Environment

(d) LunarLander Environment

Figure 6: Neuromodulator Dynamics Analysis. For each environment: (a) Temporal activation patterns of each neuromodulator (top), showing the evolution of DA, 5-HT, NE, and ACh levels over training; (b) Learning rate modulation (middle-left), demonstrating how the neuromodulators influence learning speed; (c) Attention modulation (middle-right), showing the dynamic focusing of learning; and (d) Correlation heatmap (bottom) revealing interactions between neuromodulators. The analysis demonstrates sophisticated coordination patterns, with particularly strong DA-NE coupling during critical learning phases.

ulatory systems, BIARM achieves substantial performance improvements (1.06% to 155.94%) over baseline Progressive Neural Networks while significantly reducing training variance and maintaining excellent resistance to catastrophic forgetting.

The experimental validation across four OpenAI Gymnasium environments demonstrates that coordinated adaptation consistently outperforms fixed-parameter approaches, with particularly dramatic gains in complex control tasks. The comprehensive ablation studies confirm that each adaptive module contributes distinctly to overall performance, validating the four-factor design and highlighting the importance of reward processing (DA) and knowledge integration (ACh) modules for effective sequential learning.

The framework achieves high performance retention rates (94-100%), demonstrating practical value for real-world applications requiring continual adaptation. Although the computational overhead (52-87% increase in training time per step) is substantial, the cost-benefit analysis shows that performance gains justify this investment, particularly in complex environments where coordinated knowledge integration is critical.

Future work should explore extension to larger-scale environments including visual tasks and robotics applications; investigation of non-linear neuromodulator interactions incorporating multiplicative effects and inhibition-excitation dynamics; development of more principled task novelty measures based on prediction error or entropy; exploration of learnable time constants that adapt during training; and computational optimization strategies including selective module activation and hardware acceleration to reduce computational overhead while preserving coordination benefits.

Acknowledgement

The authors gratefully acknowledge the support provided by LESIA (Laboratory of Expert Systems, Image and Signal Processing) at the University of Biskra, Algeria, for providing access to research infrastructure essential for conducting this study.

References

[1] Rusu, A. A., Rabinowitz, N. C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., & Hadsell, R. (2016). Progressive neural networks. *arXiv preprint arXiv:1606.04671*. DOI: 10.48550/arXiv.1606.04671.

[2] Hervella, Á. S., Rouco, J., Novo, J., & Ortega, M. (2024). Multi-adaptive optimization for multi-task learning with deep neural networks. *Neural Networks*, 170, 254–265. DOI: 10.1016/j.neunet.2023.11.038

[3] Boyle, N., Betts, S., & Lu, H. (2024). Monoaminergic Modulation of Learning and Cognitive Function in the Prefrontal Cortex. *Brain Sciences*, 14(9), 902. DOI: 10.3390/brainsci14090902.

[4] Mei, J., Meshkinnejad, R., & Mohsenzadeh, Y. (2023). Effects of neuromodulation-inspired mechanisms on the performance of deep neural networks in a spatial learning task. *iScience*, 26(2), 106043. DOI: 10.1016/j.isci.2023.106043.

[5] Vecoven, N., Ernst, D., Wehenkel, A., & Drion, G. (2020). Introducing neuromodulation in deep neural networks to learn adaptive behaviours. *PLOS ONE*, 15(1), e0227922. DOI: 10.1371/journal.pone.0227922.

[6] Schmidgall, S., Ziaei, R., Achterberg, J., Kirsch, L., Hajiseyedrazi, S. P., & Eshraghian, J. (2024). Brain-inspired learning in artificial neural networks: A review. *APL Machine Learning*, 2(2), 021501. DOI: 10.1063/5.0186054.

[7] Cardozo Pinto, D. F., et al. (2024). Opponent control of reinforcement by striatal dopamine and serotonin. *Nature*. DOI: 10.1038/s41586-024-08412-x.

[8] Avery, M. C., & Krichmar, J. L. (2017). Neuromodulatory systems and their interactions: A review of models, theories, and experiments. *Frontiers in Neural Circuits*, 11, 108. DOI: 10.3389/fncir.2017.00108.

[9] van de Ven, G. M., Tuytelaars, T., & Tolias, A. S. (2024). Continual learning and catastrophic forgetting. *arXiv preprint arXiv:2403.05175*. DOI: 10.48550/arXiv.2403.05175.

[10] Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., ... Hadsell, R. (2017). Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13), 3521–3526. DOI: 10.1073/pnas.1611835114.

[11] Mallya, A., Davis, D., & Lazebnik, S. (2018). PackNet: Adding multiple tasks to a single network by iterative pruning. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7765–7773. DOI: 10.1109/CVPR.2018.00805.

[12] Xu, Z., Zhou, Y., Wu, T., Ni, B., Yan, S., & Liu, X. (2020). Learn to grow: A continual structure learning framework for overcoming catastrophic forgetting. *ICLR* (2020). DOI: 10.48550/arXiv.1808.03314.

[13] Ardywibowo, R., Boluki, S., Gong, X., Wang, Z., & Qian, X. (2022). VariGrow: Variational architecture growing for task- agnostic continual learning based on Bayesian novelty. *ICML*. DOI: 10.48550/arXiv.2205.09325.

[14] Van de Ven, G. M., Siegelmann, H. T., & Tolias, A. S. (2020). Brain-inspired replay for continual learning with artificial neural networks. *Nature Communications*, 11(1), 4069. DOI: 10.1038/s41467-020-17866-2.

[15] Aljundi, R., Lin, M., Goujaud, B., & Bengio, Y. (2019). Gradient based sample selection for online continual learning. *Advances in Neural Information Processing Systems*, pp. 11816–11825. DOI: 10.48550/arXiv.1903.08671.

[16] Zenke, F., Poole, B., & Ganguli, S. (2017). Continual learning through synaptic intelligence. *Proceedings of the International Conference on Machine Learning*, PMLR, pp. 3987–3995. DOI: 10.48550/arXiv.1703.04200.

[17] Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12, 2121–2159. DOI: 10.5555/1953048.2021065.

[18] Zeiler, M. D. (2012). ADADELTA: An adaptive learning rate method. *arXiv preprint arXiv:1212.5701*. DOI: 10.48550/arXiv.1212.5701.

[19] Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. *ICLR* (2015). DOI: 10.48550/arXiv.1412.6980.

[20] Loshchilov, I., & Hutter, F. (2019). Decoupled weight decay regularization. *ICLR* (2019). DOI: 10.48550/arXiv.1711.05101.

[21] Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., & Han, J. (2019). On the variance of the adaptive learning rate and beyond. *arXiv preprint arXiv:1908.03265*. DOI: 10.48550/arXiv.1908.03265.

[22] Chen, S., Zhang, C., & Mu, H. (2024). An Adaptive Learning Rate Deep Learning Optimizer Using Long

and Short-Term Gradients Based on G–L Fractional-Order Derivative. *Neural Processing Letters*, 56, 106. DOI: 10.1007/s11063-024-11571-7.

[23] Wei, P., Shang, M., Li, J., Peng, Y., Qin, B., Li, R., & Liu, X. (2024). Efficient adaptive learning rate for convolutional neural network based on quadratic interpolation egret swarm optimization algorithm. *Heliyon*, 10(18), e37814. DOI: 10.1016/j.heliyon.2024.e37814.

[24] Shazeer, N., & Stern, M. (2018). Adafactor: Adaptive learning rates with sublinear memory cost. *arXiv preprint arXiv:1804.04235*. DOI: 10.48550/arXiv.1804.04235.

[25] FitzGerald, T. H. B., Dolan, R. J., & Friston, K. (2015). Dopamine, reward learning, and active inference. *Frontiers in Computational Neuroscience*, 9, 136. DOI: 10.3389/fncom.2015.00136.

[26] Guerrero-Criollo, R. J., Castaño-López, J. A., Hurtado-López, J., & Ramirez-Moreno, D. F. (2023). Bio-inspired neural networks for decision-making mechanisms and neuromodulation for motor control in a differential robot. *Frontiers in Neurorobotics*, 17, 1078074. DOI: 10.3389/fnbot.2023.1078074.

[27] Zhou, Y., Wang, H., Xu, S., Li, Z., Chen, Y., Wang, X., & Zhang, L. (2025). Hybrid neural networks for continual learning inspired by corticohippocampal circuits. *Nature Communications*, 16, 907. DOI: 10.1038/s41467-025-56405-9.

[28] Wang, L., Zhang, X., Su, H., & Zhu, J. (2023). Incorporating neuro-inspired adaptability for continual learning in artificial intelligence. *Nature Machine Intelligence*, 5(12), 1066–1083. DOI: 10.1038/s42256-023-00747-w.

[29] Ostapenko, O., Puscas, M., Klein, T., Jahnichen, P., & Nabi, M. (2 019). Learning to remember: A synaptic plasticity driven framework for continual learning. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11321–11329. DOI: 10.1109/CVPR.2019.01158.

[30] Zhang, Z., Zou, Y., Lai, J. and Xu, Q. (2023). M2DQN: A robust method for accelerating deep Q-learning network. *Proceedings of the International Conference on Machine Learning and Computing*, pp. 116–120.