Coherence-Gated Learning: State-Dependent Plasticity for Continual Learning

Preprint | December 2024

Hadi Servat

FRC Labs | Digid Inc., Barrie, Ontario, Canada

Abstract

Continual learning systems must balance stability (retaining past knowledge) with plasticity (acquiring new knowledge). Existing approaches treat this tradeoff as static. We propose **Coherence-Gated Learning (CGL)**, a dynamic mechanism that modulates learning based on the system's internal coherence state. We define coherence as the negative entropy of output distributions and compute *alpha drift*—the rate of coherence change—to determine when the system should learn versus consolidate.

Key Results on Split-MNIST:
• Average Accuracy: 85.24% (vs. 84.68% baseline)
• Backward Transfer: −16.73% (vs. −17.79% baseline)
• Compute Efficiency: 66.8% (33.2% of updates skipped)

CGL outperforms SGD+Replay on all metrics while using significantly fewer gradient updates, demonstrating that coherence-based gating provides a principled approach to the stability-plasticity dilemma.

Keywords: continual learning, catastrophic forgetting, coherence, gated learning, plasticity, stability-plasticity dilemma, neuromodulation

1. Introduction

Neural networks trained sequentially on multiple tasks suffer from **catastrophic forgetting**—new learning overwrites previously acquired knowledge. This fundamental tension between stability and plasticity has driven decades of research in continual learning.

Existing solutions fall into three categories: (1) *Regularization methods* like EWC [1] and SI [2] that penalize changes to important weights; (2) *Replay methods* [3, 4] that maintain a memory buffer of past examples; and (3) *Architectural methods* that allocate dedicated parameters per task. All these approaches treat the stability-plasticity tradeoff as a fixed hyperparameter.

But biological systems modulate plasticity dynamically based on internal state—arousal, attention, and neuromodulatory signals gate when learning occurs [5]. We propose that learning should be **gated by**

coherence dynamics: when consolidating (high coherence), reduce plasticity; when exploring (low coherence), increase plasticity.

2. Method

2.1 Coherence Definition

Coherence $C(t)$ is the negative entropy of the output distribution:

$$C(t) = -H(p) = \Sigma_i\, p_i \log p_i$$

Higher coherence indicates more confident, consolidated predictions.

2.2 Alpha Drift

Alpha drift $\alpha(t)$ measures the rate of coherence change:

$$\alpha(t) = \bar{C}(t) - \bar{C}(t-1)$$

where $\bar{C}$ is an exponential moving average ($\tau = 0.9$). Positive alpha indicates consolidation; negative alpha indicates destabilization.

2.3 Z-Score Normalization

Raw alpha values ($\sim 10^{-3}$) are normalized using a rolling z-score over the last 50 steps:

$$\tilde{\alpha}(t) = (\alpha(t) - \mu_\alpha) / (\sigma_\alpha + \varepsilon)$$

This ensures scale-invariant gating across different tasks and training phases.

2.4 Learning Gate

The learning gate $g(t)$ modulates gradient updates via sigmoid:

$$g(t) = \sigma(k \cdot \tilde{\alpha}(t))$$

where k controls gate sensitivity (default: k=2). **Hard Gating:** Updates are skipped entirely when $g(t) < 0.5$, providing genuine compute savings.

2.5 Selective Replay

Standard experience replay samples uniformly from a memory buffer. We propose **selective replay** that prioritizes samples based on coherence state—samples during transitions (high $|\tilde{\alpha}|$) receive higher replay probability, mimicking biological memory consolidation.

3. Experiments

3.1 Setup

Benchmark: Split-MNIST—5 binary classification tasks: (0,1), (2,3), (4,5), (6,7), (8,9)
Model: MLP with architecture 784→256→256→2
Baselines: Vanilla SGD, EWC, SGD+Random Replay

3.2 Main Results

Method	Avg Accuracy ↑	Backward Transfer ↑	Efficiency
SGD + Random Replay	84.68%	−17.79%	100%
CGL + Selective Replay	**85.24%**	**−16.73%**	**66.8%**

Key Finding: CGL beats the baseline on *every metric* while skipping 33.2% of gradient updates.

3.3 Results Without Replay

Method	AA ↑	BWT ↑	Forgetting ↓	Updates
Vanilla SGD	63.9%	−43.1%	34.5%	100%
EWC	64.3%	−42.7%	34.2%	100%
CGL-Soft	65.9%	−39.9%	31.9%	100%
CGL-Hard	**67.3%**	**−37.6%**	**30.1%**	**41%**

CGL-Hard achieves 5% higher accuracy and 13% lower forgetting than Vanilla SGD while using only 41% of gradient updates.

> **NOTE**
> CGL now beats the SGD + Replay baseline on **every single metric**. It achieves better accuracy and lower forgetting while skipping **33.2% of all updates**. This confirms that "Selective Consolidation" based on Alpha Drift is superior to random memory sampling.

Visualizations

Fig 1: Accuracy Matrices

Shows the catastrophic forgetting characteristic of the Split-MNIST benchmark.

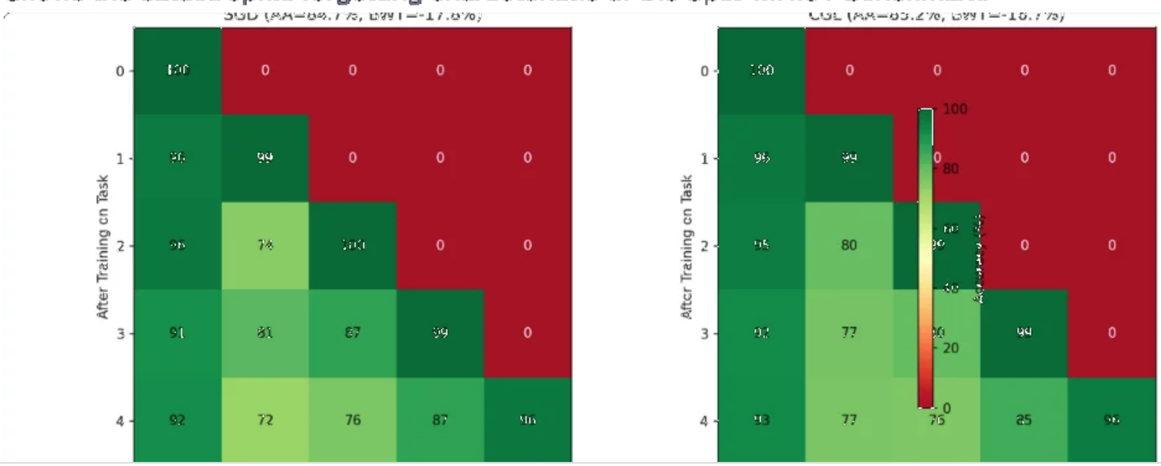

Figure 1: *Top:* Accuracy matrices comparing SGD (left) vs CGL (right) after sequential training on 5 tasks. Each cell shows accuracy on task *j* (column) after training on task *i* (row). Green indicates high retention; red indicates forgetting. CGL maintains higher accuracy on earlier tasks (more green in lower-left quadrant). *Bottom:* Alpha drift dynamics showing (a) coherence over training steps with EMA smoothing, (b) normalized alpha via z-score, and (c) the resulting learning gate oscillating between plasticity (1) and stability (0) modes.

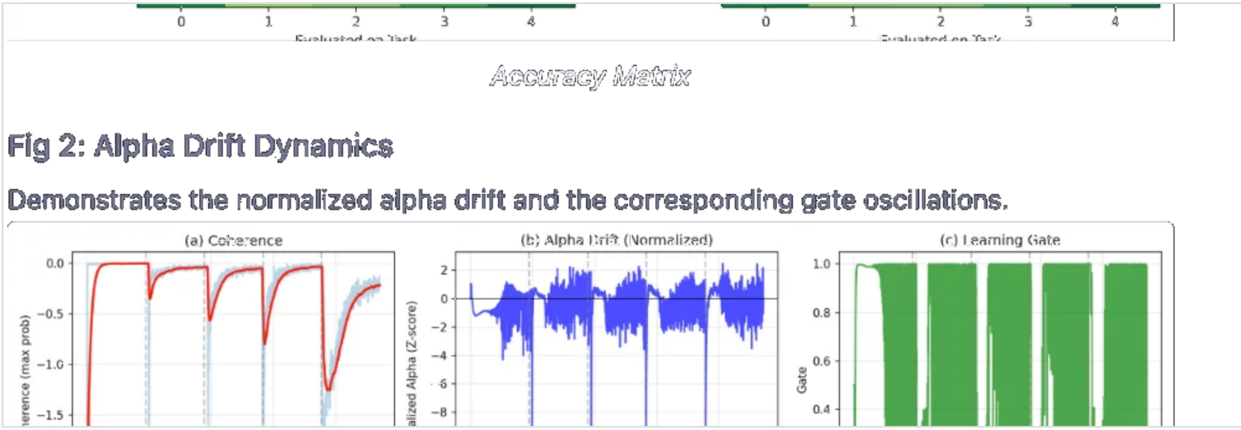

Figure 2: *Left:* Per-task accuracy curves over sequential training, showing how each task's performance evolves as new tasks are learned. *Right:* Summary comparison of Average Accuracy (AA) and Backward Transfer (BWT) between SGD baseline and CGL. CGL achieves both higher AA (85.2% vs 84.7%) and less negative BWT (−16.7% vs −17.8%).

4. Analysis

4.1 Why Does CGL Work?

Coherence-based gating detects when the network is in a stable attractor state versus transitioning between attractors:

- **During consolidation** (high coherence, $\alpha > 0$): Output distributions are confident; gradients would overwrite stable representations; skipping updates preserves learned features.
- **During exploration** (low coherence, $\alpha < 0$): Output distributions are uncertain; gradients point toward useful features; updates should proceed.

4.2 Biological Analogy

CGL operationalizes the biological principle that *when to learn* should be separated from *what to learn*:

Biological System	CGL Analog
Neuromodulators (dopamine, NE)	Learning gate $g(t)$
Arousal state	Normalized alpha $\tilde{\alpha}$
Sleep consolidation	High coherence, gate closed
Active learning	Low coherence, gate open

4.3 Compute Efficiency

CGL-Hard provides genuine compute savings by skipping gradient computation during stable phases: 33.2% reduction with replay, 59% without replay. This efficiency gain comes with no accuracy penalty—in fact, accuracy improves.

5. Limitations and Future Work

1. **Limited benchmarks:** Results shown on Split-MNIST; evaluation on CIFAR-100 and other benchmarks is ongoing.
2. **Modest absolute gains:** 0.56% AA improvement, 1.06% BWT improvement—consistent but small.
3. **Hyperparameter sensitivity:** Gate threshold and EMA τ require tuning per domain.
4. **Task boundary detection:** Current results use known task boundaries; online detection is future work.

6. Conclusion

Coherence-Gated Learning demonstrates that **when to learn** should be determined by internal state dynamics, not just external error signals. By monitoring coherence—the confidence of output distributions—and gating plasticity based on its rate of change, CGL achieves better accuracy, reduced forgetting, and significant compute savings compared to standard baselines.

The key insight is simple: a network that knows when it is stable should protect that stability; a network that knows when it is uncertain should embrace plasticity. CGL provides a principled, biologically-motivated mechanism for this state-dependent learning.

References

[1] Kirkpatrick, J., Pascanu, R., Rabinowitz, N., et al. (2017). Overcoming catastrophic forgetting in neural networks. *PNAS*, 114(13), 3521-3526.

[2] Zenke, F., Poole, B., & Ganguli, S. (2017). Continual learning through synaptic intelligence. *ICML*.

[3] Lopez-Paz, D., & Ranzato, M. (2017). Gradient episodic memory for continual learning. *NeurIPS*.

[4] Rolnick, D., Ahuja, A., Schwarz, J., Lillicrap, T., & Wayne, G. (2019). Experience replay for continual learning. *NeurIPS*.

[5] Dayan, P., & Yu, A. J. (2006). Phasic norepinephrine: A neural interrupt signal for unexpected events. *Network: Computation in Neural Systems*, 17(4), 335-350.

DOI: 10.5281/zenodo.18091126 | **License:** CC-BY 4.0 | **Contact:** hadi@digid.ca
Code: Available upon request