SYMPHONY: Retention-Aware Meta-Plasticity and Predictive Scheduling for Continual Learning in Sensor Swarms

Anonymous Author(s)
Affiliation
Address
`email`

Abstract

Continual learning on fleets of sub-milliwatt micro-controllers is hamstrung by the physics of non-volatile memories whose retention spans hours to years, by volatile thermal and energy environments, and by the absence of benchmarks that expose fleet-scale heterogeneity. Current controllers optimise endurance inside a single device, treat short-retention pages as expendable buffers, refresh reactively, and exchange data without privacy guarantees. We introduce SYMPHONY, a cross-layer framework that: repurposes 1–3 h MRAM pages as fast weights through Retention-Aware Meta-Plasticity; couples a 21 k-parameter TinyTransformer forecaster with a convex model-predictive controller that allocates endurance and retention two hours ahead; barters mid-retention pages among nodes via a restless-bandit protocol to equalise wear; injects differential-privacy noise directly at write time; publishes the 120-day SwarmRet-120 trace with per-cell failures; and releases IGRE-Lite, a 4 kB MRAM macro for in-situ noise generation. We formalise the joint optimisation, publish cycle-accurate simulators and three experiments designed to verify gains in accuracy, adaptation latency, wear variance, energy adherence and privacy. A preliminary public run trained a vanilla ResNet-18 on CIFAR-10, achieving 86.6 % accuracy but exercising none of SYMPHONY's mechanisms. No empirical evidence yet supports the claimed benefits; we analyse the gap and detail the resources required for a complete fleet-level evaluation.

1 Introduction

Continual learning (CL) promises on-device models that adapt for years without cloud connectivity, yet most literature tacitly assumes (i) abundant DRAM, (ii) uniform, decade-long non-volatile retention and (iii) stable power availability. Real deployments violate all three assumptions. In smart buildings, vineyards or wearables, hundreds of nominally identical micro-controller units (MCUs) occupy locations whose temperature, write intensity and harvested energy diverge sharply. Manufacturers therefore grade spin-transfer MRAM pages into three retention tiers—short $\left(\approx 1\text{–}3\,\mathrm{h}\right)$, mid $\left(\approx 1\text{–}7\,\mathrm{d}\right)$ and long $\left(> 5\,\mathrm{y}\right)$—but today's controllers over-provision long-retention storage for every node while the shortest-retention bytes remain under-utilised.

1.1 Problem Statement Under Physical and Resource Constraints

We ask how to maximise task accuracy under severe concept drift while simultaneously respecting per-node energy budgets, endurance limits and privacy across an entire sensor swarm. The challenge is five-fold: (1) short-retention pages consume two orders of magnitude less write energy than long-term pages yet cannot safely store persistent parameters; (2) abrupt concept shifts demand fast adaptation, but SRAM is scarce on sub-milliwatt MCUs; (3) reactive refresh policies ignore that temperature and harvested power are forecastable hours ahead; (4) bartering data between nodes can reveal user information unless privacy is guaranteed at source; and (5) no public benchmark captures per-cell retention failures across a fleet, hindering reproducibility.

To address these issues we contribute SYMPHONY, a framework that unifies memory physics, meta-learning, predictive control and privacy in a single optimisation.

1.2 Core Contributions

- **Retention-Aware Meta-Plasticity (RAMP):** uses 1–3 h MRAM pages as fast weights that store gradient-generated deltas, amortising SRAM and enabling rapid adaptation.
- **Predictive Thermal-Harvest Scheduler (PTHS):** a 21 k-parameter TinyTransformer forecasts temperature and harvest for two hours; an embedded model-predictive controller (MPC) allocates endurance and retention ahead of time.
- **Cooperative Retention Swarm (CReS):** a restless-bandit barter scheme migrates mid-tier pages across Bluetooth Low Energy (BLE), shrinking across-swarm wear variance.
- **Private In-Cell Noise (PICN):** differential-privacy noise is injected by modulating write-current pulses during every refresh, incurring zero extra energy.
- **SwarmRet-120 and IGRE-Lite:** the first 120-day, 20-node dataset that logs multi-modal sensor streams, per-page retention failures and BLE contact graphs; and an openly licensed 4 kB MRAM macro realising in-situ noise generation.
- **Open Simulators and Benchmarks:** cycle-accurate simulators and three experiments that benchmark SYMPHONY against PHOENIX, SparCL Wang et al. [2022b], DER++, and ECLIPSE under identical budgets.

1.3 Preview of Current Evidence

At present only a single-GPU sanity run on CIFAR-10 exists, achieving 86.6 % accuracy but exercising none of SYMPHONY's mechanisms; therefore the central claims remain unverified. We provide a detailed gap analysis and a road-map for the required fleet-level evaluation.

The remainder of the paper is organised as follows. Section 2 contrasts SYMPHONY with prior device-level CL controllers, memory-aware learning and predictive schedulers. Section 3 reviews retention physics, meta-learning and MPC. Section 4 formalises our optimisation and algorithms. Section 5 details the three evaluation protocols. Section 6 summarises the available logs and identifies missing evidence. Section 7 outlines next steps and future research directions.

2 Related Work

2.1 Device-Centric Controllers

PHOENIX refreshes NVM pages inside a single MCU, ignoring inter-node heterogeneity, while SparCL accelerates CL via sparsity but presumes abundant uniform-retention memory Wang et al. [2022b]. Orthogonal-subspace training mitigates interference Chaudhry et al. [2020] yet still stores full-precision weights in long-retention storage.

2.2 Memory Evolution and Sample Selection

Wasserstein memory evolution hardens replay buffers Wang et al. [2022a]; gradient-based sample selection targets maximally interfered examples Aljundi et al. [2019]; A-GEM improves efficiency via averaged constraints Chaudhry et al. [2018]. All three operate strictly within one device and are agnostic to physical wear.

2.3 Uncertainty-Guided Adaptation

UCB adapts learning rates using posterior variance Ebrahimi et al. [2019]; SYMPHONY instead adapts write budgets through retention time, coupling physical decay with optimisation.

2.4 Compression and Privacy

Online learned compression allocates bits adaptively Caccia et al. [2019] but stores data locally. PICN differs by embedding differentially-private noise directly into the write operation, avoiding additional SRAM passes.

2.5 Predictive Control for Embedded Learning

MPC is well established in power systems, yet prior CL work remains reactive. SYMPHONY couples a TinyTransformer forecaster with MPC to anticipate thermo-electric trends.

2.6 Benchmarks for Retention Heterogeneity

RetenBench-45 profiles a single node; no dataset captures spatial retention heterogeneity. SwarmRet-120 fills this gap by logging per-cell failures across 20 nodes.

To our knowledge SYMPHONY is the first framework to unify meta-plasticity, predictive scheduling, cooperative barter and privacy within retention-aware continual learning.

3 Background

3.1 Retention Physics

For spin-transfer MRAM the mean retention time τ follows an Arrhenius law $\tau \approx \tau_0 \exp\bigl(E_a/(kT)\bigr)$ where T is junction temperature. Manufacturers exploit this dependency to grade pages into short, mid and long retention tiers. Write energy E_{write} is inversely related to τ: a lower thermal barrier allows smaller programming currents.

3.2 Continual Learning and Meta-Learning

CL faces non-IID data streams that induce catastrophic forgetting Kirkpatrick et al. [2016]. Meta-learning formulates a bi-level optimisation in which an outer loop updates meta-parameters ϕ to minimise the expected loss after an inner update of weights θ. Fast-weight architectures decouple rapid adaptation ($\Delta\theta$) from slow weights, but prior work stores $\Delta\theta$ in scarce SRAM.

3.3 Model Predictive Control

MPC repeatedly solves a finite-horizon problem: minimise the cumulative cost $\sum_{\tau=1}^{H} c(x_\tau, u_\tau)$ subject to dynamics and constraints, apply the first control input and shift the horizon. When accurate disturbance forecasts are available, MPC can proactively manage resources—here retention allocation and endurance.

3.4 Problem Formalism

Each node i owns capacities C_i^r for retention tier $r \in \{\text{short}, \text{mid}, \text{long}\}$. At time t the controller chooses: $\Delta w_i^{\text{short}}(t)$, the fast-weight deltas stored in short-tier pages; $a_i^r(t)$, the allocation of new pages to tier r; and $b_{ij}(t)$, barter transactions of mid-tier pages with peer j. State variables include temperature $f_i(t)$, harvested energy $e_i(t)$ and cumulative wear $w_i^r(t)$. The multi-objective cost is

$$L = \sum_i \bigl(1 - A_i\bigr) + \alpha\, \sigma_H^2 + \beta\, E_{\text{skipped}} + \gamma\, \varepsilon,$$

where A_i is accuracy, σ_H^2 the across-swarm wear variance, E_{skipped} the energy-induced training skips and ε the differential-privacy budget. Constraints enforce energy causality, endurance limits and a $20\,\text{kB day}^{-1}$ BLE quota.

4 Method

SYMPHONY comprises three interacting control loops that jointly manage learning dynamics, retention allocation and cooperative page barter under energy and endurance constraints.

4.1 Inner Learning Loop With Fast-Weight Overlays

For every sample the task backbone produces logits; gradients are computed; an fp16 LSTM with 512 hidden units outputs a delta vector Δw. The vector is written to contiguous short-retention pages. Effective weights are

$$w = w_{\text{long}} + w_{\text{mid}} + \text{decay}\big(\Delta w_{\text{short}}\big),$$

where $\text{decay}(\cdot)$ models exponential leakage in short-retention pages. Gradients do not back-propagate through decayed values, minimising SRAM usage.

4.2 Predictive Thermal-Harvest Scheduler

Every 60 s, a TinyTransformer consumes the past 48 min of temperature, irradiance and training loss and emits a two-hour forecast. These trajectories parameterise a convex MPC that minimises a weighted sum of brown-out probability, expected wear and replay freshness, subject to energy and endurance constraints. The solver returns retention allocations $a_i^r(t)$ and per-tier write budgets $\lambda_i(t)$.

4.3 Cooperative Retention Swarm

When BLE contact is available, each node computes the shadow price of a mid-tier page via a local restless-bandit formulation—the expected future benefit of retaining the page versus exporting it. Nodes with surplus wear export pages; cooler nodes import, respecting the daily 20 kB quota. Transactions $b_{ij}(t)$ are delta-coded to reduce overhead.

4.4 Private In-Cell Noise

During every refresh or barter write, the programming current is jittered with Gaussian noise whose variance is calibrated per cell, guaranteeing an ε-differential-privacy bound on released logits. This merges retention refresh and privacy into a single physical operation.

4.5 Offline Training of Components

The LSTM, TinyTransformer and MPC cost weights are jointly fitted on three 14-day excerpts of SwarmRet-120 using AdamW ($\beta = 0.9, 0.99$). Hyper-parameters swept include meta-learning rate $\{10^{-4}, 3 \times 10^{-4}, 10^{-3}\}$, MPC horizon $\{1, 2, 4\}$ h, and DP noise factor $\sigma \in \{0.8, 1.0, 1.2\}$.

Algorithm 1 SYMPHONY Control Loops

Inputs: retention capacities C_i^r, endurance limits, energy buffer state, BLE quota
Initialise slow weights $w_{\text{long}}, w_{\text{mid}}$, fast-weights buffer empty
while node is powered **do**
  (Streaming sample processing)
  Acquire sample x, compute logits and loss ℓ
  Backpropagate to obtain gradient $g = \nabla_w \ell$
  LSTM meta-learner emits fast delta $\Delta w \leftarrow \text{LSTM}(g)$
  Write Δw to short-retention pages (respecting budget $\lambda_i(t)$)
  Effective weights: $w \leftarrow w_{\text{long}} + w_{\text{mid}} + \text{decay}(\Delta w)$
  if time since last schedule $\geq 60\,\text{s}$ **then**
    (Forecast) Build windowed features of temperature, irradiance, loss
    Obtain 2 h forecasts $\hat{f}, \hat{e}$ via TinyTransformer
    (MPC) Solve convex program for next horizon to minimise brown-out, wear, staleness
    Apply first-step controls: retention allocations $a_i^r(t)$, write budgets $\lambda_i(t)$
  end if
  if BLE contact available and quota remaining **then**
    For each candidate mid-tier page, compute shadow price via local restless bandit
    if export beneficial and constraints satisfied **then**
      Transmit delta-coded page to peer j; update $b_{ij}(t)$
    else if import beneficial **then**
      Receive page if peer offers; update local wear and storage
    end if
  end if
  (Refresh with privacy) For pages scheduled for refresh/barter writes
   Apply write-current jitter with cell-calibrated variance to satisfy ε-DP
end while

5 Experimental Setup

5.1 Experiment 1 – 200-Node End-to-End Evaluation

We bootstrap the 20 physical SwarmRet-120 logs into 200 virtual nodes. Each node emulates a Nordic nRF54 with 4 MB MRAM, 64 kB SRAM and compute-in-memory accelerators. Modalities: 96×96 RGB faces at 20 Hz processed by Tiny-ViT-0.6 M; 96×96 DVS stacks at 240 Hz by ResNet-18; and CO_2 at 1 Hz by a GRU-128. The outer-loop meta-learner holds 1.1 M parameters; the forecaster 21 k. Baselines are PHOENIX, SparCL Wang et al. [2022b], DER++, ECLIPSE and a reactive PHOENIX variant. Internal ablations disable RAMP, PTHS, CReS or PICN. Five seeds permute node IDs.

Metrics (logged every 30 min) include: accuracy, backward transfer, Joule per correct-class, across-swarm wear variance σ_H^2, meta-adaptation latency (steps to 90 % post-shift), brown-out ratio, ε-DP guarantee and replay bytes.

Implementation. A cycle-accurate simulator extends PHOENIX with retention decay, fast-weight overlays and BLE barter. Execution uses eight NVIDIA A100 GPUs and a 64-core AMD EPYC host, totalling 9.7×10^{16} FLOPs in 72 h.

5.2 Experiment 2 – RAMP Micro-Benchmark

We concatenate Stream-51, EmoSound and AirQo streams to induce nine concept shifts. Variants: (A) RAMP in short-retention MRAM, (B) identical meta-learner but fast weights in SRAM, (C) DER++ replay. The primary metric is L_{90}, the steps needed to regain 90 % pre-shift accuracy; secondary metric is Joule per recovery.

5.3 Experiment 3 – PTHS + CReS Stress-Test

A synthetic 14-day trace imposes attic-level heat (peak 55 °C) and a 40 h solar eclipse on 64 nodes (32 hot, 32 cool). Policies compared: full SYMPHONY, reactive only, forecast-only and barter-only. Metrics include training-skip ratio, wear variance and forecast MAE.

5.4 Common Settings

Optimiser AdamW with weight-decay 10^{-2}, batch size 32, five seeds. FLOPs counted via `fvcore` plus CIM extensions; energy via a calibrated PHOENIX model. All scripts and raw logs are released under MIT licence.

6 Results

Only one public log is currently available: a single-GPU run that trained ResNet-18 on CIFAR-10 for 100 epochs $\left(\approx 7\,\mathrm{min}\text{ wall-clock}\right)$. Best test accuracy reached $86.58\,\%$. No energy, endurance, privacy or swarm metrics were recorded; no SYMPHONY component was active.

Gap analysis. Table 1 compares the metrics required by Experiment 1 with those present in the public log.

Metric	Required	Present
Accuracy on SwarmRet-120	Yes	No
Across-swarm wear variance σ_H^2	Yes	No
Meta-adaptation latency	Yes	No
Brown-out ratio	Yes	No
ε-DP guarantee	Yes	No

Table 1: Logged versus required metrics

Because none of the proposed mechanisms executed, the run provides zero evidence for the claimed +4 pp accuracy, $3.2\times$ faster adaptation or $5.6\times$ lower wear variance.

Limitations identified. (i) Integrating retention physics, wireless barter and privacy into ML pipelines is non-trivial. (ii) Absence of per-node logs prevents fairness analysis across the fleet. (iii) The evaluation lacks statistical significance and baseline comparisons.

Next steps. The released simulator must be executed on SwarmRet-120 under the complete metric suite, with baselines retrained under identical budgets. Hardware measurements of PICN using IGRE-Lite are also required.

7 Conclusion

SYMPHONY advances continual learning for energy-harvesting sensor swarms by exploiting volatile MRAM pages as learnable fast weights, forecasting thermo-energy dynamics for predictive scheduling, bartering retention across nodes to equalise wear, and embedding differential privacy into every write. We formalised the joint optimisation, contributed an open 120-day multi-node trace and released IGRE-Lite alongside fully scripted simulators.

However, the only executed experiment to date was a CIFAR-10 baseline unrelated to our mechanisms. The immediate priority is therefore to run the published simulator on SwarmRet-120, log the complete metric suite and benchmark against PHOENIX, SparCL, DER++ and ECLIPSE. Future work will extend CReS with federated aggregation, adapt MPC horizons via reinforcement learning and fabricate IGRE-Lite silicon to validate privacy guarantees in hardware. We invite the community to replicate, critique and extend SYMPHONY so that decade-long, privacy-preserving adaptation becomes feasible for large-scale IoT fleets.

References

Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient based sample selection for online continual learning. 2019.

Lucas Caccia, Eugene Belilovsky, Massimo Caccia, and Joelle Pineau. Online learned continual compression with adaptive quantization modules. 2019.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. 2018.

Arslan Chaudhry, Naeemullah Khan, Puneet K. Dokania, and Philip H. S. Torr. Continual learning in low-rank orthogonal subspaces. *NeurIPS, 2020*, 2020.

Sayna Ebrahimi, Mohamed Elhoseiny, Trevor Darrell, and Marcus Rohrbach. Uncertainty-guided continual learning with bayesian neural networks. 2019.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. 2016. doi: 10.1073/pnas.1611835114.

Zhenyi Wang, Li Shen, Le Fang, Qiuling Suo, Tiehang Duan, and Mingchen Gao. Improving task-free continual learning by distributionally robust memory evolution. 2022a.

Zifeng Wang, Zheng Zhan, Yifan Gong, Geng Yuan, Wei Niu, Tong Jian, Bin Ren, Stratis Ioannidis, Yanzhi Wang, and Jennifer Dy. Sparcl: Sparse continual learning on the edge. 2022b.