Self-Organizing Resilience for Long-Term Threat Adaptation in Neuromorphic Space Systems

Sylvester Kaczmarek
Department of Computing
Imperial College London
London, UK
research@sylvesterkaczmarek.com

***Abstract*—Static defenses are brittle against the non-stationary threats common in long-duration space missions. We propose a framework for self-organizing resilience where a Spiking Neural Network (SNN) dynamically adapts its own structure to counter novel adversarial tactics. Governed by an information-theoretic objective that balances representational fidelity against computational cost, the network autonomously grows or prunes neural populations to specialize against previously unseen threat signatures. We present preliminary results from a cislunar gateway case study where the adaptive SNN is subjected to a low-rate data injection attack designed to evade static detectors. The adaptive model successfully learned the new threat pattern, reducing per-window inference time by over 40% compared to its static counterpart, with no degradation in nominal performance. We provide explicit triggers, a two-stage commit with rollback, and an audit log, treating online adaptation as a security control bounded by runtime envelopes.**

I. Introduction

Autonomous systems in space must operate for years in contested environments where threat landscapes evolve.

A. The Brittleness of Static Defenses in Long-Duration Missions

Adversaries can develop new tactics, and unforeseen system-environment interactions can create novel failure modes.

Security systems based on static, pre-trained detectors are inherently brittle in this context. Their performance degrades as the operational environment drifts away from their training distribution, a phenomenon known as concept drift [10]. For a multi-year mission, a detector trained on pre-launch data will inevitably become obsolete, leaving the system vulnerable to threats it was not designed to recognize [2], [24].

This challenge is particularly acute for machine learning-based monitors. While they offer powerful detection capabilities, their fixed architectures and parameters represent a static defense. An adversary who can identify a blind spot in the model's representation can exploit it indefinitely. The long communication latencies and limited bandwidth of space missions make frequent retraining from the ground impractical. Using an average Earth–Moon distance of 238,855 miles and a signal propagation speed of about 186,000 miles/s yields approximately 2.6 seconds round-trip for cislunar assets [27], [28], and Mars one-way latency can range from 4 to 24 minutes (8 to 48 minutes round-trip), precluding interactive debugging [28]. Furthermore, constrained contact windows and oversubscribed deep-space networks limit the data volume available for model updates [29], [30]. True long-term autonomy requires systems that can adapt their own defensive posture in-situ, without human intervention [14].

B. Position on Adaptation as a First-Class Security Control

We argue that online structural adaptation should be treated not as a mere convenience of machine learning, but as a first-class security control [4], [5]. A system that can dynamically reconfigure its own internal structure in response to novel threats offers a powerful form of resilience. Instead of relying on a fixed set of defenses, such a system can autonomously specialize its resources to counter emerging adversarial tactics.

This paper presents a framework for achieving such self-organizing resilience in Spiking Neural Networks (SNNs) [6]. We reframe the concept of adaptive morphology, typically used for improving learning efficiency, as a security mechanism. By governing the network's structural changes with a principled, information-theoretic objective, we can guide its adaptation in a stable and verifiable manner. This allows the network to grow new neural populations to represent and detect previously unseen attacks, effectively patching its own perceptual blind spots in real-time [2].

This approach is framed within a defense-in-depth strategy. We assume an adversary operating post-decryption, for instance on a compromised data bus, where traditional cryptographic integrity checks are insufficient. While model-based estimators like Extended Kalman Filters (EKFs) can detect deviations from expected dynamics, they can be evaded by the kind of stealthy injections that preserve local statistics. Our framework is designed to provide a layer of semantic integrity checking to complement these existing defenses.

C. Contributions

This paper makes the following contributions:

Workshop on Security of Space and Satellite Systems (SpaceSec) 2026
23 February 2026, San Diego, CA, USA
ISBN 978-1-970672-02-2
https://dx.doi.org/10.14722/spacesec.2026.23019
www.ndss-symposium.org

- We present a framework for self-organizing resilience in SNNs, where structural adaptation is governed by an information-theoretic objective.
- We demonstrate through a case study that this adaptive approach can successfully counter a novel, stealthy cyberattack that evades a static baseline model.
- We provide preliminary results quantifying the performance gains of the adaptive system in detecting the novel threat.
- We propose a governance model for this capability, including runtime assurance guards and auditable interfaces to ensure the safety and predictability of the adaptive process.

II. A Framework for Self-Organizing Resilience

Our approach is based on an information-theoretic adaptive morphology framework [31]. This framework recasts the problem of structural plasticity, or a network's ability to change its own structure, as a formal optimization problem. This provides a principled basis for guiding the network's self-organization, moving beyond heuristic rules to a verifiable, objective-driven process. The goal is to create a system that can autonomously and safely specialize its internal representations to counter novel threats as they emerge during a long-duration mission.

A. Information-Theoretic Objective for Adaptation

The core principle of the framework is that the network's structure, defined by its graph of neurons and synaptic weights W, should adapt to maximize an objective function $\mathcal{J}$ [13], [1]. This function formalizes the trade-off between two competing goals: preserving mission-critical information about the input and minimizing the consumption of limited onboard resources. We define the objective as:

$$\mathcal{J} = I(X;Y) - \lambda C(W), \tag{1}$$

where $I(X;Y)$ is the mutual information between the input sensor data X and the network's internal representation Y [12]. This term quantifies how much information about the environment is successfully captured by the network. In practice, direct computation of mutual information is intractable. We therefore estimate it using a variational lower bound, specifically a noise-contrastive estimator over a buffer of recent flight data [12], [1]. Maximizing this term drives the network to learn faithful and discriminative representations, making it more difficult for an adversary to craft a perturbation that is both small and effective [23].

The second term, $C(W)$, measures the structural complexity of the network, which serves as a proxy for resource consumption. For our purposes, we define it as the ℓ_0 norm of the weight matrix, which is equivalent to the total count of active synapses. The trade-off parameter $\lambda > 0$ is a hyperparameter that maps to a flight budget on computational and power resources. A higher λ penalizes complexity more heavily, encouraging the network to find sparser, more energy-efficient solutions.

B. Principled Rules for Structural Plasticity

The mechanisms for adding and removing neurons are derived as discrete operations that approximate a gradient ascent on the objective function $\mathcal{J}$ [7]. This ensures that structural changes are not random or heuristic, but are principled steps intended to improve the network's overall performance against the objective.

Neuron Growth (Threat Specialization): A new neural population is grown when the system detects a persistent failure to represent a subset of the input data. This is formally triggered when the local mutual information for a specific cluster of inputs (e.g., data associated with a persistent, low-confidence alert) falls below a predefined threshold. This condition signals a representational bottleneck or a defensive blind spot. The new neurons are then initialized to be sensitive to the statistical features of this poorly represented data, effectively creating a new, specialized unit to detect this specific class of threat. This process is analogous to a targeted basis expansion in functional approximation, designed to reduce the local representational error [23].

Neuron Pruning (Resource Reallocation): Neurons are marked for removal based on their saliency, which is a measure of their contribution to the overall objective function $\mathcal{J}$. The saliency of a neuron is approximated by the magnitude of the gradient of the objective with respect to that neuron's activity. Neurons with low saliency are those that contribute little to the network's information processing and can be considered redundant. By periodically pruning these underutilized neurons, the framework reallocates computational and memory resources to where they are most needed, ensuring that the network remains efficient and focused on the most relevant threats while operating within its resource budget.

III. Threat Model and Assumptions

The adversary's goal is to compromise the integrity of the autonomous system's perception, either to cause a mission failure or to mask a secondary malicious activity. We assume a capable adversary who has achieved a foothold on the spacecraft's data bus, allowing them to manipulate sensor or telemetry streams before they reach the neuromorphic monitor [24], [9].

The adversary can inject, delay, and replay data streams, including low-rate patterns intended to steer the system's online adaptation. They can also induce partial sensing faults and timing jitter consistent with the mission profile. However, we assume the adversary cannot modify the runtime assurance shell, the adaptation policy code itself, or the protected non-volatile memory where rollback checkpoints are stored [4], [5]. We also assume that radiation-induced faults occur with rates bounded by mission analysis, and that nominal operation constitutes the vast majority of the mission time. The goal of our defense is to bound false alarms, latency, and energy consumption while improving the time-to-detect for previously unseen adversarial tactics.

A. *Formalizing the Low-Rate Data Injection Attack*

To test the adaptive capabilities of our framework, we design a novel threat that is specifically intended to evade static, non-adaptive detectors. This "low-and-slow" or low-rate data injection attack is a form of stealthy data manipulation [9]. The adversary's objective is to introduce a malicious bias into a sensor stream without triggering detectors that rely on first-order (mean) or second-order (variance) statistical checks [10].

Let $X = \{x_1, x_2, \ldots, x_T\}$ be a time-series of nominal sensor readings from a distribution P_{nom}. The adversary crafts a perturbed stream $X' = \{x'_1, x'_2, \ldots, x'_T\}$ by adding a small, time-varying perturbation δ_t. The attack is designed to satisfy two constraints. First, over any short time window W, the local statistics of the perturbed stream match the nominal distribution:

$$\mathbb{E}_{t \in W}[X'] \approx \mathbb{E}[X] \quad \text{and} \quad \text{Var}_{t \in W}[X'] \approx \text{Var}[X] \qquad (2)$$

Second, a malicious bias is introduced very slowly over a long period, such that the cumulative effect becomes significant. This can be modeled as a slow drift in a hidden parameter. Detecting this attack requires a model that can identify subtle violations of higher-order statistical properties or long-term temporal dependencies, making it an ideal test case for an adaptive defense [11].

IV. Adaptation to a Novel Stealth Attack

To evaluate the framework, we use a high-fidelity simulation of a Lunar Gateway-class space station [20].

A. *Experimental Setup for Lunar Gateway Monitoring*

The SNN monitors N_{ch} telemetry channels (sampled at Δt) windowed into W-second segments.

Channels include: (i) thermal sensors from the habitation module, (ii) main power bus voltage/current, and (iii) communications link SNR. This multi-modal stream tests whether the network learns cross-system correlations (e.g., thermal and power couplings) rather than single-channel thresholds. The system is initially trained on a wide variety of known anomaly types.

B. *Low-rate data injection threat*

We use the attack defined in Section III, matching first- and second-order windowed statistics while inducing slow drift in a hidden parameter.

C. *Models for Comparison*

We compare the performance of two models:

- **Adaptive SNN:** The full proposed framework, with the adaptation mechanism enabled, allowing it to change its structure online.
- **Static SNN:** An SNN with an identical initial architecture, but with the adaptation framework disabled [6], [15]. Its structure is fixed after initial training.

V. Preliminary Results

The static SNN shows persistent false negatives and a drop in $\widehat{I}(X;Y)$.

TABLE I
Performance Comparison on the Novel Low-Rate Attack.

Metric	**Adaptive SNN**	**Static SNN**
F1-Score (Novel Attack)	**0.72 ± 0.08**	0.61 ± 0.09
Per-window Inference Time (ms)	**2.1 ± 0.7**	3.5 ± 1.1
Energy/Inference (μJ)	65.1 ± 9.2	62.8 ± 8.9
Nominal Set F1	0.87 ± 0.03	0.87 ± 0.03

A. *Evidence of Structural Adaptation to the Novel Threat*

When the trigger is exceeded, the system grows ≈ 50 neurons targeted to the low-information cluster and refines them via spike-timing-dependent plasticity (STDP) [8].

Stabilization completes within 35–55 minutes. This matches intent: identify a blind spot, specialize, and resume detection under envelopes.

B. *Quantifying the Performance Gain in Threat Detection*

The structural adaptation led to a significant and measurable improvement in security performance. Table I presents a comparison of the key performance metrics for the two models, evaluated on the novel attack after the adaptive model had stabilized. The Static SNN's failure is evident in its low F1-score, as its fixed representations were unable to distinguish the subtle, higher-order correlations of the attack from nominal noise [2]. The Adaptive SNN, by creating new, specialized representations, was able to successfully learn the threat pattern. Stabilization of the new structure and its performance completed within 35–55 minutes after the initial growth event. During the extended simulation, 23% of all adaptation attempts were automatically rolled back by the runtime envelopes for failing to meet performance or resource constraints, and no control-loop deadline misses relative to $L_{\max}$ were recorded.

The Adaptive SNN achieves a higher F1-score on the novel attack (0.72 vs. 0.61) and reduces per-window inference time by over 40% compared to the static model, while preserving nominal performance (Nominal Set F1 unchanged), indicating no catastrophic forgetting [23]. Stabilization completes within 35–55 minutes; during this period, runtime envelopes and existing monitors remain active (Section VI).

C. *Analysis of Resource Overheads*

Resource impact is small: energy per inference rises by 3.7% and latency remains under $L_{\max}$ [4]. Envelopes roll back 23% of attempted updates that would violate bounds, and no control-loop deadlines are missed.

VI. Runtime Envelopes and Commit Policy

Flight systems must adapt within verifiable limits to avoid instability and manipulation [4], [5], [25]. We enforce explicit envelopes on false alarms, latency, and energy; a two-stage commit with rollback; and an auditable log of every structural change [22].

Envelopes and commit condition. We require

$$\text{FPR}_{t+1} \le \alpha, \quad L_{t+1} \le L_{\max}, \quad E_{t+1} \le E_{\max}, \quad \Delta \ge \delta$$

evaluated on a protected nominal buffer and a fixed library of attack-like stimuli.

Poisoning guard Accept updates only if corroborated by at least two independent channels or if the trigger persists for a dwell time d; otherwise reject the proposal regardless of envelope checks.

A. *Fencing Adaptation with Budgets, Monitors, and Rollback Policies*

The adaptation process is fenced by hard budgets and runtime envelopes (Table II), with checkpointed rollback to a known-good configuration in protected non-volatile memory [4], [5], [22]. Table I reports per-window compute time; $L_{\max}$ is the hard deadline enforced by the control loop.

B. *Interfaces for Operator Audits and Trust*

To build operator trust, the adaptation process must be transparent and auditable [17], [18].

The system generates an auditable log per adaptation event capturing the trigger, the structural diff, and the measured outcome, enabling operator review and post-hoc verification [19], [25].

VII. Adaptation Event Procedure

We handle each adaptation event with a minimal, auditable sequence:

1) Trigger. Raise a candidate when the information gain signal exceeds threshold $I_t = \widehat{I}(X;Y)_t - \widehat{I}(X;Y)_{\text{nom}} \geq \tau$ with dwell time d, and corroboration across channels when available.

2) Propose. Allocate a growth proposal of Δn neurons (or a pruning set) targeted to the low-information cluster. Initialize with sensitivity to the suspect features; freeze unrelated pathways.

3) Sandbox. Run on the shadow path for k windows, updating only the proposed subgraph.

4) Check. Evaluate envelopes $\{\alpha, L_{\max}, E_{\max}\}$ and compute detection gain Δ.

5) Decide. Commit if bounds hold and $\Delta \geq \delta$ [22]. Otherwise rollback and log the outcome. Enforce cooldown c.

6) Log. Store $\langle t, I_t, \text{FPR}_t, L_t, E_t, \text{action}\rangle$ and a compact hash of the structural diff for audit.

VIII. Flight Constraints and Policy Mapping

Budgets and timers are set from spacecraft power, compute, storage, and contact constraints, summarized in Table II [4].

IX. Limitations and Failure Modes

This approach does not eliminate poisoning risk; it limits its effect through corroboration, dwell time, and budgets [24]. Shadow-path evaluation can underfit tactics that manifest only under closed-loop actuation. Envelope thresholds require mission-specific tuning; overly strict bounds can block beneficial updates. We have not proven global stability of repeated structural updates; instead we provide bounded-risk operation through rollback and budgets [4], [5]. Once the growth budget is reached, the policy must shift to reallocation (prune low-saliency neurons to free capacity) or reject updates and escalate for ground review if no safe reallocation is available. Results here are limited to a single low-rate injection scenario and a short horizon with one dominant growth event; longer-horizon behavior under repeated adaptations remains future work.

TABLE II
Runtime envelopes and budgets used in experiments

Parameter	Value	Rationale
α (FPR bound)	0.6%	Nominal alert load for ops
$L_{\max}$ (latency)	1.5 ms	Control-loop deadline
$E_{\max}$ (per inference)	70 μJ	Power headroom margin
$N_{\max}$ (neurons)	20,000	Hard growth cap
$S_{\max}$ (synapses)	2×10^6	Hard growth cap
k (shadow windows)	5	Transient smoothing
d (dwell time)	90 s	Contact-aligned review
c (cooldown)	10 min	Prevent thrash
δ (min gain)	+0.05 F1	Commit threshold
λ_{day} (budget)	2 attempts	Limit exposure

X. Related Work and Novelty

Adaptive SNNs, such as evolving SNNs (eSNNs), and other online learners address distribution shift for accuracy, often without the strict flight constraints or auditability required for space missions [6], [7], [26]. Runtime assurance shells fence actuator commands but rarely govern structural learning [4], [5], [22]. Spaceborne security work emphasizes detection and protocol hardening rather than bounded structural adaptation [14], [11]. Our contribution is to treat morphology change as a security control governed by envelopes on false alarms, latency, and energy, with a two-stage commit and audit trail suited to long missions [4].

XI. Conclusion and Future Work

This paper presented a framework for self-organizing resilience, where a neuromorphic system can autonomously adapt its own structure to counter novel threats. Our preliminary results show that this approach can successfully defend against a stealthy attack that evades a static detector, with minimal overhead. By fencing the adaptation process with clear governance rules and providing auditable logs, we can ensure the safety and predictability of this powerful capability. This work reframes online adaptation as a first-class security control, offering a path toward truly resilient, long-term autonomy for contested space environments [14], [19].

Future work will focus on testing the framework against more sophisticated, adaptive adversaries and on developing formal verification methods for the runtime assurance guards [25], [5]. Planning for an in-orbit demonstrator to validate this technology in a real space environment is also a key next step [3], [21].

References

[1] A. A. Alemi, I. Fischer, J. V. Dillon, and K. Murphy, “Deep Variational Information Bottleneck,” in *International Conference on Learning Representations (ICLR)*, 2017.

[2] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards Deep Learning Models Resistant to Adversarial Attacks," in *International Conference on Learning Representations (ICLR)*, 2018.

[3] D. Izzo, A. Hadjiivanov, D. Dold, G. Meoni, and E. Blazquez, "Neuromorphic computing and sensing in space," *arXiv preprint* arXiv:2212.05236, 2022.

[4] J. D. Schierman, M. D. DeVore, N. D. Richards, and M. A. Clark, "Runtime Assurance for Autonomous Aerospace Systems," *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 12, pp. 2205–2217, 2020.

[5] D. Cofer, I. Amundson, R. Sattigeri, A. Passi, C. Boggs, E. Smith, L. Gilham, T. Byun, and S. Rayadurgam, "Run-Time Assurance for Learning-Enabled Systems," in *Proc. NASA Formal Methods (NFM)*, Lecture Notes in Computer Science, vol. 12229, 2020, pp. 361–368.

[6] W. Maass, "Networks of Spiking Neurons: The Third Generation of Neural Network Models," *Neural Networks*, vol. 10, no. 9, pp. 1659–1671, 1997.

[7] E. O. Neftci, H. Mostafa, and F. Zenke, "Surrogate Gradient Learning in Spiking Neural Networks: Bringing the Power of Gradient-Based Optimization to SNNs," *IEEE Signal Processing Magazine*, vol. 36, no. 6, pp. 51–63, 2019.

[8] J. P. Pfister and W. Gerstner, "Triplets of spikes in a model of spike timing-dependent plasticity," *Journal of Neuroscience*, vol. 26, no. 38, pp. 9673–9682, 2006.

[9] T. Wu, X. Wang, S. Qiao, X. Xian, Y. Liu, and L. Zhang, "Small Perturbations are Enough: Adversarial Attacks on Time Series Prediction," *Information Sciences*, vol. 587, pp. 794–812, 2022.

[10] G. Pang, C. Shen, L. Cao, and A. van den Hengel, "Deep Learning for Anomaly Detection: A Review," *ACM Computing Surveys*, vol. 54, no. 2, Article 38, 2021.

[11] G. Hundman, V. Constantinou, C. Laporte, I. Colwell, and T. Soderstrom, "Detecting Spacecraft Anomalies Using LSTMs and Nonparametric Dynamic Thresholding," in *Proc. 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 387–395.

[12] B. Poole, S. Ozair, A. van den Oord, A. A. Alemi, and G. Tucker, "On Variational Bounds of Mutual Information," in *Proc. 36th International Conference on Machine Learning (ICML)*, 2019, pp. 5171–5180.

[13] N. Tishby, F. C. Pereira, and W. Bialek, "The Information Bottleneck Method," in *Proc. 37th Annual Allerton Conference on Communication, Control, and Computing*, 2000.

[14] I. A. D. Nesnas, L. M. Fesq, and R. A. Volpe, "Autonomy for Space Robots: Past, Present, and Future," *Current Robotics Reports*, vol. 2, no. 3, pp. 251–263, 2021.

[15] M. Davies et al., "Loihi: A Neuromorphic Manycore Processor with On-Chip Learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, 2018.

[16] S. Sharmin, N. Rathi, P. Panda, and K. Roy, "Inherent Adversarial Robustness of Deep Spiking Neural Networks: Effects of Discrete Input Encoding and Non-Linear Activations," in *Computer Vision – ECCV 2020*, 2020, pp. 399–414.

[17] J. D. Lee and K. A. See, "Trust in Automation: Designing for Appropriate Reliance," *Human Factors*, vol. 46, no. 1, pp. 50–80, 2004.

[18] M. Mitchell, S. Wu, A. Zaldivar, P. Barnes, L. Vasserman, B. Hutchinson, E. Spitzer, I. D. Raji, and T. Gebru, "Model cards for model reporting," in *Proc. Conference on Fairness, Accountability, and Transparency (FAT* '19)*, 2019, pp. 220–229.

[19] NASA Science Mission Directorate, "Artificial Intelligence Workshop Report," Tech. Rep., 2024.

[20] M. S. Anderson, "Gateway at the Crossroads of Sustainable Lunar Exploration: Program Overview and Architecture," NASA, Tech. Rep. NTRS 20230013256, 2023.

[21] C. D. Schuman, S. R. Kulkarni, M. Parsa, J. P. Mitchell, P. Date, and B. Kay, "Opportunities for neuromorphic computing algorithms and applications," *Nature Computational Science*, vol. 2, pp. 10–19, 2022.

[22] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control Barrier Function Based Quadratic Programs for Safety Critical Systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2017.

[23] A. Achille and S. Soatto, "Information Dropout: Learning Optimal Representations Through Noisy Computation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 12, pp. 2897–2905, 2018.

[24] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and Harnessing Adversarial Examples," in *International Conference on Learning Representations (ICLR)*, 2015.

[25] S. A. Seshia, D. Sadigh, and S. S. Sastry, "Formal Methods for Semi-Autonomous Driving," in *Proc. 52nd Annu. Design Automation Conf. (DAC)*, 2015, pp. 148:1–148:5.

[26] S. Schliebs and N. Kasabov, "Evolving spiking neural network-a survey," *Evolving Systems*, vol. 4, no. 2, pp. 87–98, 2013. doi: 10.1007/s12530-013-9074-9.

[27] NASA Glenn Research Center, "Seeing the Earth, Moon, and Sun to Scale," NASA Glenn K-12 (Numbers), educational webpage.

[28] K. Schauer, "Space Communications: 7 Things You Need to Know," NASA, Oct. 6, 2020.

[29] NASA Office of Inspector General, "Audit of NASA's Deep Space Network," Report No. IG-23-016, July 12, 2023.

[30] Jet Propulsion Laboratory (California Institute of Technology), "Deep Space Network Services Catalog," DSN No. 820-100, Rev. H, Issue Date: June 6, 2022.

[31] S. Kaczmarek, "Axiomatic Foundations of Trustworthy Anomaly Detection in Cislunar Autonomy," *The Journal of Space Philosophy*, vol. 14, no. 2, Fall 2025.