

# Neuraxon

## A New Neural Growth & Computation Blueprint

David Vivancos\* vivancos@vivancos.com

and Jose Sanchez\*\* jose.sanchezgarcia@unir.net

\*Artificiology Research; \*\*UNIR University ; (\*&\*\*) Qubic Science

**Abstract.** We introduce *Neuraxon*, a novel computational unit inspired by biological neurons yet extending beyond classic perceptrons through *trinary logic* (1, 0, -1). This design captures excitatory, neutral, and inhibitory dynamics, incorporates continuous processing in which inputs never cease, and makes output timing a first-class variable. Computation occurs at both neuron and synapse levels, with plasticity realised via synaptic collapse, formation, and reconnection—plus the rare, but possible, neuron death. By modelling fast and slow receptor channels, autoreceptors, and self-modulation, Neuraxons sustain spontaneous firing and intrinsic activity, mirroring task-irrelevant yet persistent processes seen in brains. We also present a Neuraxon-based architecture based on biological plausibility, small-world connectivity yields scale-free synchronisation and self-organising feedback loops (“Matrioska-doll” motifs), promoting rapid adaptation to non-stationary environments. These findings highlight potential for continuous, real-time learning and cognitive flexibility, paving the way for more energy-efficient and robust AI systems. For a deeper exploration we hybridize Neuraxon with Qubic’s Aigarth Intelligent Tissue as a first case study.

**Keywords:** Neuraxon · Trinary states · Continuous processing · Synaptic dynamics · Neural plasticity · Spontaneous activity · Temporal synchronisation · Bio-inspired computation

### 1 Neuraxon: A Trinary Bioinspired Neural Unit

Neuraxon uses trinary states (1, 0, -1) for excitatory, neutral, and inhibitory activity, processing continuous inputs with dynamic internal signaling. Beyond mere excitation/inhibition, it incorporates a third synaptic state that modulates the postsynaptic membrane without triggering an immediate spike. These intermediate signals act as neuromodulators and subtly shift the neuron’s electrical state and future responsiveness, rather than directly depolarizing or hyperpolarizing it.

For example, many neurotransmitters act through G-protein-coupled metabotropic receptors whose activation induces slower changes in the neuron: rather than opening an ion channel directly, they initiate cascades of second messengers that can close or open ion channels at a distance and alter cell’s excitability [1].

Unlike ionotropic receptors (for rapid excitatory or inhibitory responses), metabotropic receptors generate slower and longer-lasting synaptic effects, adjusting the membrane potential and the neuron's response to future inputs [1]. Thus, neuromodulators such as dopamine, serotonin, or acetylcholine do not “switch on” or “off” the neuron directly, but rather tune its sensitivity and firing patterns. These changes include the activation or closure of specific ion channels (e.g., potassium or calcium channels influenced by intracellular messengers), modifying the firing threshold or the spontaneous firing rate. In sum, the modulatory signal creates an intermediate state in the neuron: it does not cause it to fire, but it prepares or predisposes it to respond differently to subsequent signals.

A key facet of this third synaptic state is subthreshold inputs and “silent” synapses. Subthreshold inputs—stimuli too weak alone to trigger an action potential—still leave a membrane “trace”: small depolarizations accumulate across synapses, summing spatially or temporally. Coincident subthreshold signals can bring the membrane closer to threshold, so a slight extra excitation suffices to fire. Even without spiking, they modulate internal dynamics. For instance, during synchronized cortical “Up” states (slow-wave sleep), subthreshold-activated synapses weaken if not paired with postsynaptic firing [2]. Thus, subthreshold inputs form a functional intermediate state that influences both synaptic strength and a neuron's future likelihood of firing.

Silent synapses are a key example of the third synaptic state: anatomically present but functionally inactive under resting conditions. Typically found in the hippocampus, they have NMDA receptors but lack functional AMPA receptors [3]. Thus, when glutamate is released, it binds to NMDA receptors, which remain non-conductive due to  $Mg^{2+}$  block at resting potential. Without concurrent depolarization, the synapse stays “silent”. During long-term potentiation (LTP), silent synapses are “unsilenced” by inserting AMPA receptors, becoming active.

Dendritic branches display a third, intermediate signaling state via local dendritic spikes—small, branch-restricted action potentials mediated by NMDA or  $Na^+/Ca^{2+}$  conductances that seldom elicit somatic firing [4]. These spikes nonlinearly amplify and integrate inputs on their branch, generating a transient depolarizing plateau that boosts responsiveness and triggers localized  $Ca^{2+}$  influx essential for plasticity. Distal dendritic spikes are necessary for LTP at distant synapses: when paired with synaptic input, the calcium surge activates cascades that strengthen those connections [4]. Thus, neurons can “learn” without a global spike—the dendritic spike serves as a modulatory third state, enabling branch-specific plasticity and dendritic autonomy.

This third synaptic state also enriches Hebbian learning by requiring modulatory signals beyond mere coactivation. For example, dopamine acting at D1 metabotropic receptors “opens the gate” for hippocampal STDP—without dopamine, identical spike timing fails to change synapses [5]. Likewise, acetylcholine, norepinephrine, and serotonin adjust synaptic thresholds and gains across networks, reshaping circuit dynamics (e.g., sleep/wake transitions, selective attention) and essential to consolidate LTP or LTD.

Therefore the “third synaptic state” thus refers to all these subtle forms of influencing neuronal activity without directly triggering an immediate electrical impulse and sustained by diverse neurobiological mechanisms (metabotropic receptors, local dendritic spikes, silent synapses, among others). It determines how signals are integrated in dendrites, how plastic changes are triggered, and how neurons adjust their behavior according to neurochemical context. Far from being mere “background noise,” they enable neurons not only to transmit information but to filter it, store it latently, and weigh it according to global relevance. It adds an additional layer of complexity and flexibility to neural circuits, reflecting a fundamental principle of brain organization that computational models are now beginning to incorporate.

Neuromodulators in the brain may resemble metaplasticity, rarely considered in existing spiking (SNNs) and nonspiking artificial neural networks (ANNs). Reported efficient brain-inspired computing algorithms whereby the long-term synaptic potentiation and depression are modified in a nonlinear manner depending on the neuromodulator level showed promising results [6, 7]. Neuraxon, a neuromodulation-inspired mechanism driven by neuromodulators can enhance ANN performance with greater flexibility, robustness, and adaptability [8, 9].

## 2 Time, Speed, and Continuous Flow

Time is fundamental in biological neural processing, with neurons integrating inputs over time and action potentials encoding critical information [10]. While traditional artificial networks use discrete time steps, the Neuraxon model introduced here employs continuous processing where inputs flow as a constant stream and timing shapes outputs, mirroring biological neural activity with an internal state that evolves over time through external inputs and intrinsic mechanisms, similar to membrane potential in biological neurons [11]. This can be represented by a differential equation inspired by continuous-time recurrent neural networks [12].

$$\tau \frac{ds}{dt} = -s + \sum_i w_i \cdot f(s_i) + I_{\text{ext}}(t)$$

Here,  $s(t)$  represents the internal state at time  $t$ ,  $\tau$  is a time constant governing the decay rate,  $w_i$  are synaptic weights,  $f(s_i)$  is the activation function of presynaptic Neuraxons, and  $I_{\text{ext}}(t)$  denotes external inputs. This formulation ensures that the state evolves smoothly, reflecting the continuous flow of information, a principle also leveraged in recent spiking neural network (SNN) models for time-series forecasting [13]. The output of a Neuraxon is trinary—1 (excitatory), 0 (neutral), or -1 (inhibitory)—determined by applying thresholds to  $s(t)$ :

$$\text{output}(t) = \begin{cases} 1 & \text{if } s(t) > \theta_1 \\ -1 & \text{if } s(t) < \theta_2 \\ 0 & \text{otherwise} \end{cases}$$

where  $\theta_1$  and  $\theta_2$  are threshold parameters. This trinary logic, detailed in Section 1, enriches temporal dynamics by allowing the duration, frequency, and sequence of states to encode information, supporting complex signal processing akin to biological neurons [14]. Similarly, [13] demonstrate how SNNs encode continuous time-series data into spike trains, capturing temporal dependencies with high fidelity, which aligns with Neuraxon's approach to processing unending input streams. Neuraxons enables real-time pattern detection through continuous processing, unlike discrete models. As shown by [13], this approach aligns with SNNs' success in time-series forecasting and maintains spontaneous activity similar to biological brains [15]. [16] T-RevSNN complements Neuraxon by optimizing memory and energy use. Continuous flow allows real-time synaptic adjustment, offering advantages including: immediate adaptation to inputs [13], reduced processing overhead [16], enhanced temporal pattern recognition [14], and closer alignment with biological neural dynamics [11], providing a framework for adaptive intelligent systems applicable to robotics and embodied AI development.

### 3 Integrated Neuron-Synapse Computation

Unlike conventional neural networks with static weights, the Neuraxon model embeds dynamic computational capabilities within synapses, reflecting biological neural systems where synapses actively contribute to processing. Each synapse features three weights capturing different temporal dynamics: fast weight ( $w_{\text{fast}}$ ), slow weight ( $w_{\text{slow}}$ ), and metabotropic weight ( $w_{\text{meta}}$ ). Fast and slow weights mimic ionotropic receptors like AMPA and NMDA, contributing to the postsynaptic potential through rapid currents [20]. The metabotropic weight models slower modulatory influences over extended timescales. In 'modulatory' synapses,  $w_{\text{meta}}$  generates signals that alter the postsynaptic Neuraxon's behavior, such as changing firing thresholds or adaptation rates. For example, the effective threshold for transitioning to an excitatory state (1) might be adjusted as:

$$\theta_{\text{eff}} = \theta_1 - k \cdot \sum_i w_{\text{meta},i}$$

where the summation is over all incoming modulatory synapses, and  $k$  is a scaling factor. This mechanism reflects the action of biological neuromodulators like dopamine or serotonin, enabling the network to tune neuronal excitability based on ongoing activity patterns. Unlike traditional models, Neuraxon synapses are not passive connectors but active computational units with evolving states. Each synapse maintains three dynamic weights:  $w_{\text{fast}}(t)$ ,  $w_{\text{slow}}(t)$ , and  $w_{\text{meta}}(t)$ , each governed by their respective differential equations:

$$\begin{aligned} \tau_{\text{fast}} \frac{dw_{\text{fast}}}{dt} &= -w_{\text{fast}} + h_{\text{fast}}(s_i(t)) \\ \tau_{\text{slow}} \frac{dw_{\text{slow}}}{dt} &= -w_{\text{slow}} + h_{\text{slow}}(s_i(t)) \\ \tau_{\text{meta}} \frac{dw_{\text{meta}}}{dt} &= -w_{\text{meta}} + g(\text{pre\_state}, \text{post\_state}, \text{global\_signals}) \end{aligned}$$

Here,  $\tau_{\text{fast}} < \tau_{\text{slow}} < \tau_{\text{meta}}$  reflect their distinct temporal scales, with  $\tau_{\text{meta}}$  being significantly larger to capture the 'ultraslow' nature of neuromodulation effects. The functions  $h_{\text{fast}}$  and  $h_{\text{slow}}$  modulate the fast and slow weights based on the presynaptic Neuraxon's state  $s_i(t)$ , while  $g$  integrates presynaptic and postsynaptic activity, potentially alongside global modulatory signals, over prolonged periods. This formulation enables synapses to temporally filter inputs across multiple timescales, a critical feature for processing continuous, time-dependent signals [10]. The synaptic input to the postsynaptic Neuraxon is primarily driven by the fast and slow weights:

$$\text{synaptic\_input}_i(t) = (w_{\text{fast},i}(t) + w_{\text{slow},i}(t)) \cdot f(s_i(t))$$

Meanwhile,  $w_{\text{meta},i}(t)$  modulates the postsynaptic neuron's properties, such as its thresholds or adaptation rates, rather than directly contributing to the synaptic input. The postsynaptic Neuraxon integrates these synaptic inputs to update its state  $s(t)$ , following a continuous-time dynamics model [12]:

$$\tau \frac{ds}{dt} = -s + \sum_i \text{synaptic\_input}_i(t) + I_{\text{ext}}(t)$$

where  $\tau$  is the neuronal time constant, and  $I_{\text{ext}}(t)$  represents external inputs. The trinary output (1, 0, or -1) is determined by thresholding  $s(t)$ , as outlined in Section 2. This integration captures the history of synaptic activity, enabling the neuron to exhibit complex temporal behaviors like oscillations, essential for memory and pattern recognition [14].

Recent SNN research supports Neuraxon's principles. [17] introduced GRSNN, which uses synaptic delays to encode relational information in graph-based reasoning tasks, transforming the temporal domain into an additional processing dimension. This approach outperforms methods using only synaptic weights, with 20-fold energy efficiency improvements while maintaining competitive performance.

Neuraxon features plasticity at both synaptic and neuronal levels. The three weights adapt through mechanisms similar to spike-timing-dependent plasticity (STDP) [18], while structural plasticity mirrors biological turnover to optimize network efficiency [19]. The tripartite synaptic framework with  $w_{\text{meta}}$  enables simulation of complex phenomena like neuromodulation and multi-timescale information integration, aligning with biological neural systems.

## 4 Plasticity and Adaptation

These are cornerstone features of biological neural systems, enabling learning, environmental responsiveness, and functional optimization. Our model emulates biological neural properties through synaptic and neuronal mechanisms for continuous adaptation, keeping in mind that maintaining long-term plasticity presents challenges as neural networks can lose adaptability over time.

Synaptic plasticity occurs through continuous weight evolution, mirroring spike-timing-dependent plasticity in biological neurons [18]. Within Neuraxon's trinary framework—representing excitatory (1), neutral (0), and inhibitory (-1) states—specific plasticity rules apply: synapses strengthen when presynaptic state 1 precedes postsynaptic state 1, and weaken with presynaptic 1 followed by postsynaptic -1. The neutral state provides nuanced control through weight stabilization. Continuous processing enables real-time adjustments without discrete training phases, allowing integration of inputs over time for detecting complex temporal patterns [10]. To formalize this plasticity, we introduce a synaptic weight update rule that captures the temporal variation of pre- and postsynaptic activity, incorporating Hebbian principles and associativity among adjacent synapses. The change in synaptic weight,  $\Delta w$ , is defined as:

$$\begin{aligned} \Delta w = & \eta \int [A^+(t) \delta(\text{pre} = 1, \text{post} = 1) - A^-(t) \delta(\text{pre} = 1, \text{post} = -1)] dt \\ & + C + \alpha_j^{\text{adj}} \sum d_{ij} \Delta w_j \end{aligned}$$

where  $\eta$  is the learning rate,  $A^+(t)$  and  $A^-(t)$  are temporal kernels for potentiation and depression, respectively,  $\delta(\text{condition})$  are indicator functions that activate based on the trinary states (1, 0, -1) of the pre- and postsynaptic Neuraxons,  $C$  is a constant baseline adjustment, and the final term represents associativity, promoting coordinated plasticity among nearby synapses.

This equation drives continuous evolution of synaptic weights in response to Neuraxon activity patterns. The temporal kernels  $A^+(t)$  and  $A^-(t)$  prioritize recent activity, mirroring the timing windows observed in biological STDP [10]. The indicator functions enforce Hebbian learning: for instance, when both pre- and postsynaptic Neuraxons are in state 1 (excitation),  $\delta(\text{pre} = 1, \text{post} = 1)$  triggers potentiation; if the presynaptic is in state 1 and the postsynaptic in state -1 (inhibition),  $\delta(\text{pre} = 1, \text{post} = -1)$  induces depression. The neutral state (0) can stabilize weights or adjust plasticity thresholds. The constant  $C$  may model intrinsic weight drift or stabilization, while the associativity term,  $\alpha \sum_{j \text{ adjacent to } i} \frac{\Delta w_j}{d_{ij}}$ , reflects biological phenomena where plasticity in one synapse influences neighbors, weighted by proximity. This can foster synaptic clusters, enhancing associative memory storage and retrieval.

For real-time implementation, this update is expressed as a differential equation:

$$\begin{aligned} \frac{dw}{dt} = & \eta [A^+(t) \cdot \delta(\text{pre} = 1, \text{post} = 1) - A^-(t) \cdot \delta(\text{pre} = 1, \text{post} = -1)] \\ & + C + \alpha \sum_{j \text{ adjacent to } i} \frac{w_j - w_i}{d_{ij}} \end{aligned}$$

The model employs continuous weight adjustments for real-time processing in robotics and signal applications. Structural plasticity mechanisms optimize topology through synapse formation/collapse and occasional neuron death based

on activity patterns [19]. Research by [21] identifies key plasticity challenges: weight saturation, early data overfitting, and architectural rigidity. In continual learning, [22] show standard methods suffer catastrophic forgetting when learning new tasks. [23] neuroplastic expansion approach offers complementary strategies through elastic topology generation, dormant neuron pruning, and experience review for neuron consolidation. This methodology parallels existing structural plasticity mechanisms while suggesting enhancements. The trinary logic system provides advantages through its neutral state (0), which serves as a buffer enabling rapid transitions and helping prevent weight saturation. Future research could explore dynamic growth strategies and investigate how the trinary framework balances stability with adaptability.

## 5 Beyond Binary: Complex Signaling

Biological neural networks utilize complex signaling beyond binary frameworks, with inputs modulated by timing, neurotransmitter diversity, and receptor dynamics [10]. The trinary state model reflects this: excitatory (1), inhibitory (-1), and neutral (0) signals. The neutral state mirrors subthreshold inputs in biological systems [3], allowing responsiveness without immediate action while enabling swift transitions based on subsequent inputs. Continuous processing integrates inputs as constant streams, with timing directly influencing outputs similar to biological spike timing [14]. Multiple synaptic channels—fast and slow components—emulate diverse temporal dynamics of receptors like AMPA and NMDA [20], differentiating transient from sustained patterns. This capability is essential for speech recognition and motor control where temporal nuance matters. Spontaneous firing and intrinsic activity maintain network stability and facilitate learning [15]. The trinary framework enables nuanced decision-making by integrating various inputs, similar to biological neurons balancing excitatory and inhibitory potentials. This approach enhances computational sophistication and energy efficiency by keeping neurons in low-energy, poised states. Continuous processing also allows modeling of temporal phenomena like synaptic facilitation and depression [20]. Recent advances support these principles: [14] BitNet uses ternary weights (-1, 0, 1) to match full-precision transformers while reducing latency, memory usage, and energy consumption. Similarly, [17] GRSNN leverages synaptic delays for temporal processing in graph reasoning tasks, achieving competitive performance with approximately 20 $\times$  energy savings through sparse computation. These developments demonstrate the shift from binary constraints toward biologically-inspired signaling paradigms for advanced, efficient artificial intelligence.

## 6 Self-Generated Activity

Neuronal activity persists continuously, even without external stimuli, and constitutes most of the brain's energy use, playing a fundamental role in cognition

[24, 25]. This continuous background activity is key to self-learning, plasticity, and adaptability.

This spontaneous activity exhibits rich spatiotemporal structure across scales, from local stereotyped firing patterns to large-scale networks operating near criticality with scale-free topology and balanced excitation/inhibition [26].

Analyses of single-cell spike trains reveal that intrinsic firing is non-Poissonian and history-dependent: each neuron’s autocorrelation function (ACF) displays a characteristic “autocorrelation window,” i.e. the timespan over which past spikes affect current firing probability [27]. Longer windows imply greater intrinsic memory, crucial for sustaining endogenous rhythms and state-dependent modulation of inputs.

These spontaneous dynamics provide the substrate for synaptic plasticity, learning, adaptive behavior, and the emergence of a coherent self [28, 29]. The rest-stimulus interaction underscores how the pre-stimulus state, shaped by intrinsic timescales and the autocorrelation window, modulates perception and memory encoding [25].

Inspired by these principles, modern artificial networks may use spontaneous-like dynamics to enhance memory capacity and flexibility. Embedding “endogenous noise” and controllable autocorrelation windows may endow these AI models with continuous learning, resilience to perturbations, and energy-efficient temporal processing, thereby mirroring the advantages of the brain’s dynamic repertoire.

Overall, embedding spontaneous neural dynamics yields more flexible and adaptive AI. The self-driven activity keeps networks “on” and ready to integrate new information. For example, the generative-replay model successfully prevents forgetting without any external data storage [30] and the plastic network with internal recall can instantaneously restructure knowledge from its own activity patterns [31]. Conventional ANNs lack this internal engine. Mimicking the brain’s intrinsic dynamics through spiking activity, recurrent noise, and internal replay can make AI systems more efficient at continuous learning and real-time processing. In sum, self-generated neural activity in AI models offers a powerful route to brain-like adaptability, enabling ongoing learning from the network’s own spontaneous dynamics.

## 7 Time-Centric Synchronization and Future Outlook

Neuraxon’s emphasis on precise timing and oscillatory nesting fosters small-world synchronization via self-organized loops. The brain integrates across temporal and spatial scales through neuronal oscillations—from infraslow ( $< 0.1$  Hz) to gamma ( $> 30$  Hz)—each with distinct functions. The Temporo-Spatial Theory of Consciousness (TTC) posits that high-frequency rhythms nest within slower cycles via cross-frequency coupling (CFC), enabling multiscale integration and continuity of experience. TTC highlights the brain’s intrinsic temporo-spatial structure rather than specific firing patterns. Although neuroscience often probes responses to external stimuli, most cerebral energy supports spontaneous

activity, which exhibits complex temporo-spatial fluctuations and network organization, shaping processing and the birth of consciousness [32][33][34]. The brain has intrinsic neural timescales determining information integration durations. These vary across cortical regions, corresponding to different oscillatory frequencies. Historically, Neural Correlates of Consciousness (NCC) research focused on faster rhythms like alpha (8–12 Hz) and gamma (> 30 Hz), which increase during conscious perception. Slower frequencies (infraslow, 0.01–0.1 Hz; slow, 0.1–1 Hz) were often overlooked, partly as their power rises during reduced consciousness states like deep sleep. Fast and slow frequencies have distinct roles. High-frequency oscillations relate to specific conscious contents, qualifying as NCCs: their modulation correlates with perceiving stimuli or cognitive operations, sufficiently representing given conscious content. Slow frequencies, however, don't encode content but set necessary conditions for consciousness. As a large-scale temporal framework, they organize and modulate high-frequency activity. These slow rhythms are Neural Predispositions of Consciousness (NPC): necessary, but not solely sufficient, for consciousness (and its fast-rhythm-carried contents) to arise. Intact slow oscillations are vital for maintaining overall conscious state level [35, 36]. Temporal scales in the brain interact through cross-frequency coupling (CFC), where one band modulates another. A key CFC form is phase-amplitude coupling (PAC): the phase of a slow oscillation gates the amplitude of faster rhythms [37, 28]. Slow cycles create temporal windows, determining when and how strongly high-frequency bursts occur [35]. Consequently, rapid information (carried by high-frequency amplitude) is embedded within slower cycles. Combining the roles of fast and slow rhythms via CFC yields temporo-spatial nestedness—the hierarchical organization where faster processes nest within slower ones for example, gamma-band amplitudes align with theta-phase windows [35][36][39]. This nested architecture enables temporo-spatial integration: binding events across scales into a coherent whole [32]. Future research will integrate energy efficiency and specialized hardware, mirroring natural neural constraints for scalable, robust AI solutions beyond conventional perceptrons. Incorporating “nestedness” into Artificial General Intelligence (AGI) models is crucial for achieving human-like understanding and adaptability. AGI architectures should feature hierarchical structures where information processing occurs across multiple, nested temporal and complexity scales, mirroring brain-inspired designs [40]. This allows for the integration of context over varying granularities, from immediate sensory data to long-term goals. Implementation could leverage advancements in nested machine learning algorithms and bilevel optimization [41], enabling different AI components to operate and coordinate dynamically. Such an approach would foster more robust, contextually aware, and efficient embodied AI systems capable of handling complex, real-world scenarios by processing information in a deeply integrated, multi-scale fashion.

## 8 Hybridizing Neuraxon with Aigarth Intelligent Tissue

In this section, we propose a hybridization of Neuraxon—a trinary, continuous-processing neural unit inspired by biological neurons—with Aigarth, an evolutionary framework for intelligent tissue units. This integration creates a superior bioinspired neural model that combines Neuraxon’s sophisticated synaptic dynamics and temporal processing with Aigarth’s mechanisms for self-modification, mutation, and natural selection. The result is a system capable of autonomous evolution, lifelong learning, and emergent complexity, addressing limitations in traditional neural networks such as catastrophic forgetting and static architectures. We discuss how these two systems fit together, the mechanisms of hybridization, and the advantages for bioinspired AI, supported by recent advancements in neuroevolution and hybrid neural models.

Aigarth Intelligent Tissue (AIT), is designed as a foundational material for constructing AI modules that evolve through continuous self modification and selection of the fittest instances, emulating biological evolution [1]. At its core, Aigarth employs Intelligent Tissue Units (ITUs), which are circular arrangements of neurons using balanced ternary logic (-1, 0, +1) for states and weights. This ternary representation aligns with biological excitatory, neutral, and inhibitory signals, allowing for nuanced computations beyond binary models. In Aigarth, neurons are initialized with random ternary weights for a single forward input, exploring spherical structures that evolve via mutation. Mutation involves randomly selecting and adjusting a neuron’s input weight by +1 or -1, clamped to ternary values. If a weight change leads to an overflow (e.g., beyond 1 or -1), it triggers neuron spawning: a new neuron is cloned from a neighboring "spawn model" neuron and inserted adjacent in the circle. Conversely, neurons with all zero weights are removed if not part of input/output groups, optimizing the network topology. This process resembles biological neurogenesis and apoptosis, where neurons form, prune or die based on activity and utility [42]. Feedforward computation in Aigarth occurs in cycles, where each neuron computes its next state based on weighted sums from backward and forward groups in the circle, incorporating input skew for directional bias. The cycle continues until output neurons stabilize or a tick cap is reached, producing a ternary output vector. Reflection mechanisms encode and decode external data, enabling the system to "reflect" on inputs for learning. Aigarth’s evolutionary aspect is evident in its training paradigm: ITUs mutate across "training seasons" (datasets) and episodes, with version history tracking progress. This allows for selection of fittest ITUs, fostering problem-solving capabilities through Darwinian principles. Recent neuroevolution studies highlight similar benefits, where evolutionary algorithms optimize neural architectures for robustness in dynamic environments [42]. Synapses in Neuraxon feature fast, slow, and metabotropic weights, mimicking ionotropic and metabotropic receptors for multi-timescale integration [20]. Plasticity includes synaptic collapse/formation and rare neuron death, while spontaneous firing sustains intrinsic activity. This design promotes real-time adaptation and cognitive flexibility, as seen in biologically inspired heterogeneous networks [43].

To implement neuromodulation in Neuraxon Aigarth, we have selected specific parameters for activation thresholds based on the biological affinities observed in the key neuromodulators: dopamine (DA), serotonin (5HT), acetylcholine (Ach) and noradrenaline (NA). This choice is biologically justified in the need to emulate the brain's dual phasic-tonic dynamics. Low thresholds (nM, high affinity) require little amount of neurotransmitter to activate the receptor, allowing sensitive responses to low and sustained basal concentrations, which implies prolonged and energy-efficient modulation for processes such as stable reinforcement learning (DA in reward circuits) or continuous emotional regulation (5HT in the cortex). High thresholds ( $\mu\text{M}$ , low affinity) need large amount of neurotransmitter to activate, responding to intense and transient synaptic peaks. It implies a selective activation that filters weak signals and thus prevents constant overexcitation. It serves for rapid responses to salient events. For example, we incorporate thresholds in the nM range for high-affinity subtypes (1–10 nM in 5-HTA, 10–100 nM for NA), and  $\mu\text{M}$  for those of medium-low affinity ( $\text{EC}_{50} \sim 1\text{--}100$  for Ach or 1–5  $\mu\text{M}$  for DA), within our trinary model to represent the neutral/modulated state (0). This variability of the thresholds reflects the biological evolution for an optimal dynamic range, where high affinities (nM) support tonic modulation for stability and homeostasis, while low ones ( $\mu\text{M}$ ) allow phasic modulation for reactivity. There is evidence in this sense for dopaminergic receptors [52], serotonin in depression [53], tonic and burst stimulation of the locus coeruleus for noradrenaline [54], or regulation of emotions by dopamine and acetylcholine [55]. To advance in Neuraxon-Aigarth, the initial implementation starts from a single subtype per neuromodulator (D1 for DA, 5-HT1A for 5HT, M1 for Ach and  $\beta 1$  for NA). In this way the model evaluates basic robustness in simple tasks. In an initial phase, models described in the scientific literature typically recommend using around 100 neurons with 5–10 synaptic connections each when working with limited computational power. For simulations exploring synchronization, plasticity, oscillatory dynamics, and clustering, a network size between 500 and 1,000 neurons is generally advised [56]. In a second phase, different subtypes are added (D1 and D2, to balance motivation and inhibitory control;  $\alpha 2$  and  $\beta 1$ , for regulation of alertness and stress; 5-HT1A, 5-HT2A and 5-HT4 for impulsivity, cognitive flexibility and memory; M1 and M2, for cognition and autoinhibition). In this way multitask conflict resolution and "general" behaviors inspired by biological networks are emulated. The parameters optimize the model's trinary behavior incorporating synaptic dynamics such as  $w_{\text{fast}}$ ,  $w_{\text{slow}}$ ,  $w_{\text{meta}}$  to emulate ionotropic and metabotropic receptors, continuous plasticity via STDP-like rules, and self-generated activity for intrinsic learning and adaptability in non-stationary environments.

The core idea of the hybridization is to replace Aigarth's simple ternary neurons with Neuraxons, creating "Neuraxon-ITUs" where the circle consists of Neuraxons. In this hybrid, each Neuraxon in the circle inherits Aigarth's input weighting and skew but incorporates its own fast/slow/meta dynamics. Mutation

extends to Neuraxon parameters: weight adjustments apply to fast/slow/meta components differentially, with probabilities biased toward fast weights for rapid adaptation and meta for long-term modulation. Spawning clones a Neuraxon, preserving its internal state and synaptic history, while removal targets low-health Neuraxons (based on activity metrics). Feedforward computation integrates continuous time: instead of discrete ticks, the hybrid runs simulations over time intervals, with Neuraxon states evolving continuously. Reflection now involves temporal patterns, where inputs are streamed over time, and outputs are sequences of trinary states. This aligns with spiking neural network hybrids guided by neuroevolution, achieving efficient training [44]. Evolution occurs at multiple levels: individual Neuraxon parameters evolve via local plasticity, while the ITU structure evolves through Aigarth's mutation/selection. For selection, fitness is evaluated via reflection accuracy on datasets, incorporating temporal metrics like synchronization [45]. This multi-level evolution mirrors biological neural development, where genetic evolution shapes architectures and synaptic plasticity refines function [42]. To formalize, consider the hybrid state update:

$$\frac{ds_j}{dt} = -s_j + \sum_{i \in \text{circle}} (w_{\text{fast},ij} + w_{\text{slow},ij}) \cdot f(s_i) + w_{\text{meta},ij} \cdot g(\text{global})$$

Mutation probabilistically alters weights or spawns, with rates modulated by meta weights for self-regulated evolution. This fit creates a "living" neural tissue: Neuraxons provide biological fidelity, Aigarth adds evolvability. Hybrid models using bio-inspired metaheuristics have shown superior intrusion detection, demonstrating robustness [49].

### Advantages of the Hybrid Model

The hybrid offers several improvements over standalone systems. First, enhanced adaptability: Aigarth's evolution mitigates Neuraxon's potential plasticity loss [21], enabling lifelong learning without forgetting [46]. Second, emergent complexity: Spontaneous activity in Neuraxons, combined with evolutionary pruning, leads to scale-free networks with small-world properties, promoting synchronization [47]. Third, energy efficiency: Ternary logic reduces computational overhead, as in bitnet models [14], while evolution optimizes sparse structures. Fourth, better bioinspiration: The model captures evolutionary timescales, from synaptic (Neuraxon) to structural (Aigarth), aligning with temporo-spatial theories [32]. Empirical advantages are supported by hybrid SNNs with evolutionary channels, excelling in pattern recognition [45]. In robotics, such systems enable motor control modulation [51].

## 9 Base Neuraxon Architecture and Comparison

Following is the pseudo code for the Neural units (not the hybrid model), basic architecture and a comparison between other models:

**Algorithm 1** Neuraxon Units

---

```

1: function SYNAPSE(pre_id, post_id)
2:   init  $w_{fast}, w_{slow}, w_{meta}, \tau_{fast}, \tau_{slow}, \tau_{meta},$ 
3:      $modulatory\_flag, silent\_flag$ 
4:   function COMPUTE_INPUT(pre_state)
5:     if silent_flag  $\wedge$  dendritic_spikes = 0 then return 0
6:   end if
7:   return  $(w_{fast} + w_{slow}) \cdot pre\_state$ 
8: end function
9:   function UPDATE(pre_state, post_state, global_signals)
10:    update_traces(pre_state, post_state)
11:     $w_{fast} += \frac{dt}{\tau_{fast}} \cdot (h_{fast}(pre\_state) - w_{fast})$ 
12:     $+ stdp(pre\_state, post\_state) \cdot 0.3$ 
13:     $w_{slow} += \frac{dt}{\tau_{slow}} \cdot (h_{slow}(pre\_state) - w_{slow})$ 
14:     $+ stdp(pre\_state, post\_state) \cdot 0.1$ 
15:     $w_{meta} += \frac{dt}{\tau_{meta}} \cdot (g(pre\_state, post\_state, global\_signals)$ 
16:       $- w_{meta}) + associative(global\_signals)$ 
17:    adjust_integrity(pre_state, post_state)
18: end function
19: end function
20: function NEURAXON(neuron_id)
21:   init membrane_potential, trinary_state, thresholds, adaptation,
22:     spontaneous_rate, health
23:   function UPDATE(synaptic_inputs, external_input, global_signals)
24:     dendritic_sum = nonlinear_branch_integration(synaptic_inputs)
25:     membrane_potential +=  $\frac{dt}{\tau} \cdot (-membrane\_potential$ 
26:        $+ dendritic\_sum + external\_input - adaptation + spontaneous(dt))$ 
27:     adaptation +=  $\frac{dt}{\tau_{adaptation}} \cdot (-adaptation + 0.1 \cdot trinary\_state)$ 
28:     autoreceptor +=  $\frac{dt}{\tau_{autoreceptor}} \cdot (-autoreceptor$ 
29:        $+ 0.2 \cdot trinary\_state)$ 
30:     effective_theta1 = theta1 - threshold_mod(global_signals) - 0.1  $\cdot$ 
      autoreceptor
31:     effective_theta2 = theta2 - threshold_mod(global_signals) + 0.1  $\cdot$ 
      autoreceptor
32:     if membrane_potential > effective_theta1 then
33:       trinary_state = 1
34:     else if membrane_potential < effective_theta2 then
35:       trinary_state = -1
36:     else
37:       trinary_state = 0
38:     end if
39:     update_health()
40:   end function
41: end function

```

---

**Algorithm 2** Neuraxon Neural Network System

---

```

function NEURAXONNETWORK(neuron_count, connection_probability)
2:   neurons = [Neuraxon(i) for i in range(neuron_count)]
      synapses = generate_small_world_synapses(neuron_count, connection_probability)
4:   global_signals = {dopamine : 0, serotonin : 0, acetylcholine : 0,
      neighboring_activity : 0, metabotropic_modulation : 0}
6:   oscillators = init_oscillators()
      function SIMULATE_STEP(external_inputs = {}, external_modulation =
      {}, dt = 0.1)
8:     update_global_signals(dt, external_modulation)
      update_oscillators(dt)
10:    neuron_inputs = defaultdict(list)
        for all syn ∈ synapses do
12:      neuron_inputs[syn.post_id].append(syn.compute_input(
          neurons[syn.pre_id].trinary_state))
14:    end for
        for all neu ∈ neurons do
16:          neu.update(neuron_inputs[neu.id],
            external_inputs.get(neu.id, 0) + oscillator_drive(neu),
            global_signals)
18:        end for
        for all syn ∈ synapses do
20:          syn.update(neurons[syn.pre_id].trinary_state,
            neurons[syn.post_id].trinary_state, global_signals)
        end for
24:          apply_structural_plasticity()
            energy_usage += 0.01 · count_active_neurons() · dt
26:    end function
      function APPLY_STRUCTURAL_PLASTICITY
28:    remove_synapses([s for s ∈ synapses if s.integrity < 0.1
      and chance(0.01)])
30:    remove_neurons([n for n ∈ neurons if n.health < 0.1
      and chance(0.001)])
32:    if chance(0.1) then
        add_new_synapse(random_pre(), random_post())
34:    end if
      end function
36: end function
      function NEURAXONAPPLICATION(network_size)
38:    network = NeuraxonNetwork(network_size, 0.1)
      patterns = {}
40:    function PRESENT_PATTERN(pattern, steps)
        for step ∈ range(steps) do
42:          network.simulate_step(external_inputs = pattern)
        end for
44:    end function
      function RECALL_PATTERN(name, steps)
46:        partial = mask_pattern(patterns[name].pattern, 0.7)
        for step ∈ range(steps) do
48:          network.simulate_step(external_inputs = partial)
        end for
50:    end function
      function TRAIN_SEQUENCE(sequence, repetitions)
52:        for rep ∈ range(repetitions) do
          for all pattern ∈ sequence do
54:            present_pattern(pattern, 5)
          end for
56:        end for
      end function
58: end function

```

---

**Table 1.** Comparative Analysis of Neural Network Architectures

Feature	Conventional NNs	Spiking NNs	Transformers	Neuraxon
<b>Units</b>	Perceptrons with continuous values	Binary spiking neurons (fire/no-fire)	Self-attention with continuous values	Trinary state neurons (1,0,-1)
<b>State</b>	Continuous [0,1] or [-1,1]	Binary spikes with timing	Continuous vectors with attention weights	Trinary: excitatory, neutral, inhibitory
<b>Temporal Processing</b>	Discrete steps; requires RNNs/LSTMs	Inherent in spike timing	Parallel with positional encoding	Continuous with timing as core variable
<b>Synaptic Model</b>	Static weights	Weight-based with delays	Dynamic attention weights	Multi-component fast, slow, metabotropic weights
<b>Compute</b>	At neuron level	Neuron level with timing	Across attention heads and FFNs	Distributed across neurons and synapses
<b>Learning</b>	Backpropagation	STDP and variants	Backpropagation with attention	Continuous weight evolution with trinary rules
<b>Plasticity</b>	Fixed topology	Limited connection changes	Fixed with dynamic attention	Dynamic topology, synapse formation/death
<b>Self-Activity</b>	None	Some background spiking	None without input	Spontaneous firing patterns
<b>Encoding</b>	Amplitude/rate in values	Spike timing and rate	Token embeddings and attention	States, timing, and patterns
<b>Inhibition</b>	Negative weights	Inhibitory neurons	Low attention weights	Native -1 state
<b>Biological Plausibility</b>	Low	Medium	Very low	High
<b>Complexity</b>	Simple per neuron	Moderate (spike timing)	High ( $O(n^2)$ with sequence)	Very high (multi-timescale dynamics)
<b>Memory</b>	Fixed per layer	Varies with resolution	High (attention matrices)	High (multiple weight types)
<b>Applications</b>	General ML tasks	Event-based sensing	NLP, vision, generative models	Adaptive systems, continuous learning

## Appendix 1: Source Code and Interactive Demonstration

### Source Code Repository

The complete implementation of Neuraxon is available as open-source software under the MIT License at:

<https://github.com/DavidVivancos/Neuraxon>

The repository includes:

- Core implementation (`neuraxon.py`) - Pure Python with no external dependencies
- Comprehensive documentation and usage examples
- Test suite for validation
- Network parameter presets for various configurations
- JSON serialization for saving/loading network states

**Note:** The Aigarth hybrid features described in Section 7 require compliance with the Aigarth License, which prohibits military use.

### Interactive Web Demonstration

A fully interactive 3D visualization of Neuraxon networks is publicly accessible at:

<https://huggingface.co/spaces/DavidVivancos/Neuraxon>

The demonstration provides an intuitive interface for exploring Neuraxon’s dynamics without installation requirements.

**Demonstration Features Network Configuration:** Users can configure all biologically-plausible parameters including neuron counts (input, hidden, output), connection probability, membrane time constants, firing thresholds, synaptic time scales ( $\tau_{\text{fast}}$ ,  $\tau_{\text{slow}}$ ,  $\tau_{\text{meta}}$ ), learning rates, and neuromodulator baselines. Preset configurations (minimal, balanced, highly plastic, large-scale) enable rapid experimentation.

**3D Visualization:** The network is rendered in three-dimensional space with neurons arranged in a spherical topology. Neurons are color-coded by trinary state: red for excitatory (+1), blue for inhibitory (-1), and gray for neutral (0). Synaptic connections appear as curved pathways between neurons, with line thickness representing synaptic integrity.

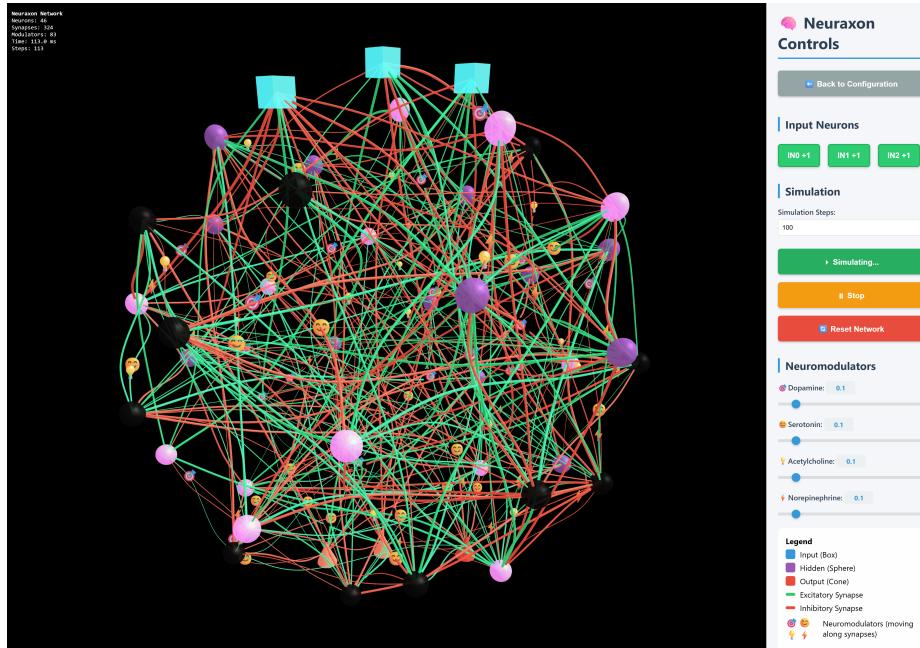
**Interactive Input Control:** Input neurons can be manually toggled between the three trinary states via button controls, allowing users to inject specific patterns and observe propagation through the network layers.

**Real-time Neuromodulation:** Four neuromodulators (dopamine, serotonin, acetylcholine, norepinephrine) are adjustable via slider controls. Visual emoji particles flow along synaptic pathways at speeds proportional to their concentration levels, providing intuitive feedback on neuromodulatory dynamics.

**Continuous Simulation:** Users can run time-stepped simulations to observe spontaneous activity, synaptic plasticity, and structural changes. The simulation displays network time, step count, and real-time state updates across all neurons.

**Technical Implementation** The demonstration employs Three.js for WebGL-based 3D rendering with orbital camera controls for interactive viewing. Neurons are represented as spherical meshes with emissive materials, while synapses utilize Bézier curves for natural connectivity visualization. Neuromodulator particles are implemented as sprite billboards with emoji textures, animated along synaptic curves using parametric interpolation. The interface updates at 60 FPS, with network state changes reflected immediately in the visual representation.

This tool serves both educational and research purposes, enabling rapid prototyping and intuitive understanding of Neuraxon's multi-scale temporal dynamics, trinary logic operations, and neuromodulatory influences on network behavior.



**Neuraxon Network Builder**

By David Vivancos & Jose Sanchez - Cubic Science

Build Your Bio-Inspired Neural Network based on Neuraxon paper & Cubic's Aigarth Intelligent Tissue

Configure all parameters using the sliders below, then click "Build Network" to create and visualize your custom Neuraxon network with trinary states, continuous processing, and complex synaptic dynamics.

Default Small Network Large Network

Sparse Dense Fast Dynamics

**Network Architecture**

Input Neurons  
Number of input neurons (1-5)  
3

Hidden Neurons  
Number of hidden neurons (1-100)  
40

Output Neurons  
Number of output neurons (1-5)  
3

Connection Probability  
Probability of synapse formation between neurons (0.0-0.2)  
0.05

**Neuron Parameters**

Membrane Time Constant (ms)  
Time constant for membrane potential decay (5.0-50.0)  
20.0

Excitatory Threshold  
Firing threshold for excitatory state (+1) (0.5-2.0)  
1.0

Inhibitory Threshold  
Firing threshold for inhibitory state (-1) (-2.0 to -0.5)  
-1.0

Adaptation Rate  
Rate of neuronal adaptation (0.0-0.2)  
0.05

Spontaneous Firing Rate  
Probability of spontaneous activity (0.0-0.1)  
0.01

Health Decay Rate  
Rate of neuron health decay without activity (0.0-0.01)  
0.001

**Synapse Parameters**

Fast Ionotropic (AMPA-like)  
Time Constant (ms)  
5.0

Weight Min/Max  
-1.0 1.0

Slow Ionotropic (NMDA-like)  
Time Constant (ms)  
50.0

Weight Min/Max  
-0.5 0.5

Metabotropic  
Time Constant (ms)  
1000.0

Weight Min/Max  
-0.3 0.3

**Neuromodulator Parameters**

Dopamine Baseline  
0.1

Serotonin Baseline  
0.1

Acetylcholine Baseline  
0.1

Norepinephrine Baseline  
0.1

Neuromodulator Decay Rate  
Rate of neuromodulator decay (0.0-0.5)  
0.1

**Simulation Parameters**

Time Step (dt) in ms  
Simulation time step (0.1-10.0)  
1.0

Simulation Steps  
Number of initial simulation steps (1-10000)  
100

**Build Neuraxon Network**

## Citation and Attribution

When using Neuraxon in research, please cite:

Vivancos, D., & Sanchez, J. (2025). Neuraxon: A New Neural Growth & Computation Blueprint. ResearchGate Preprint.

For questions or collaborations, contact:

vivancos@vivancos.com jose.sanchezgarcia@unir.net

## References

- [1] Rosenmund, C., Stern-Bach, Y., & Stevens, C. F. (2003). The tetrameric structure of a glutamate receptor channel. *Nature*, 405, 689–694. <https://journals.physiology.org/doi/full/10.1152/jn.00965.2003>
- [2] Chen, J. L., Villa, K. L., Cha, J. W., So, P. T., Kubota, Y., & Nedivi, E. (2013). Clustered dynamics of inhibitory synapses and dendritic spines in the adult neocortex. *Neuron*, 77(5), 999–1013. <https://www.nature.com/articles/nature11986>
- [3] Kerchner, G. A., & Nicoll, R. A. (2008). Silent synapses and the emergence of a postsynaptic mechanism for LTP. *Nature Reviews Neuroscience*, 9(11), 813–825.
- [4] Larkum, M. E., Nevian, T., Sandler, M., Polsky, A., & Schiller, J. (2009). Synaptic integration in tuft dendrites of layer 5 pyramidal neurons: A new unifying principle. *Science*, 325(5941), 756–760. <https://elife sciences.org/articles/01970>
- [5] Brzosko, Z., Mierau, S. B., & Paulsen, O. (2019). Neuromodulation of spike-timing-dependent plasticity: Past, present, and future. *Frontiers in Synaptic Neuroscience*, 11, 24.
- [6] Zhang, T., Cheng, X., Jia, S., Li, C. T., Poo, M. M., & Xu, B. (2023). A brain-inspired algorithm that mitigates catastrophic forgetting of artificial and spiking neural networks with low computational cost. *Science Advances*, 9(34), eadi2947. <https://doi.org/10.1126/sciadv.adi2947>
- [7] Plantec, E., Pedersen, J. W., Montero, M. L., Nisiotti, E., & Risi, S. (2024, July). Evolving Self-Assembling Neural Networks: From Spontaneous Activity to Experience-Dependent Learning. In *ALIFE 2024: Proceedings of the 2024 Artificial Life Conference*. MIT Press.
- [8] Guo, Y., Chen, Y., Liu, X., Peng, W., Zhang, Y., Huang, X., & Ma, Z. (2024, March). Ternary spike: Learning ternary spikes for spiking neural networks. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 38, No. 11, pp. 12244–12252).
- [9] Mei, J., Rodriguez-Garcia, A., Takeuchi, D., Wainstein, G., Hubig, N., Mohsenzadeh, Y., & Ramaswamy, S. (2025). Improving the adaptive and continuous learning capabilities of artificial neural networks: Lessons from multi-neuromodulatory dynamics. *arXiv preprint arXiv:2501.06762*.
- [10] Gerstner, W., Kistler, W. M., Naud, R., & Paninski, L. (2014). *Neuronal Dynamics*. Cambridge University Press.
- [11] Dayan, P., & Abbott, L. F. (2001). *Theoretical Neuroscience*. MIT Press.
- [12] Beer, R. D. (1995). On the dynamics of small continuous-time recurrent neural networks. *Adaptive Behavior*, 3(4), 469–509.

- [13] Lv, C., Zheng, X., Wang, Y., et al. (2024). Efficient and Effective Time-Series Forecasting with Spiking Neural Networks. In *Proceedings of the 41st International Conference on Machine Learning*, PMLR 235.
- [14] Ma, S., Wang, H., Ma, L., Wang, L., Wang, W., Huang, S., Dong, L., Wang, R., Xue, J., & Wei, F. (2024). BitNet: Scaling 1-bit Transformers for Large Language Models. *arXiv preprint arXiv:2402.17764*.
- [15] Deco, G., Jirsa, V. K., Robinson, P. A., Breakspear, M., & Friston, K. J. (2009). The dynamic brain: From spiking neurons to neural masses and cortical fields. *PLoS Computational Biology*, 5(8), e1000092.
- [16] Hu, J., Man, Y., Qiu, X., Chou, Y., Cai, Y., Qiao, N., Tian, Y., Xu, B., & Li, G. (2024). High-Performance Temporal Reversible Spiking Neural Networks with ... In *Proceedings of the 41st International Conference on Machine Learning*, PMLR 235.
- [17] Xiao, M., Zhu, Y., He, D., & Lin, Z. (2024). Temporal Spiking Neural Networks with Synaptic Delay for Graph Reasoning. In *Proceedings of the 41st International Conference on Machine Learning*, Vienna, Austria. PMLR 235.
- [18] Bi, G. Q., & Poo, M. M. (1998). Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type. *Journal of Neuroscience*, 18(24), 10464–10472.
- [19] Holtmaat, A., & Svoboda, K. (2009). Experience-dependent structural synaptic plasticity in the mammalian brain. *Nature Reviews Neuroscience*, 10(9), 647–658.
- [20] Citri, A., & Malenka, R. C. (2008). Synaptic plasticity: Multiple forms, functions, and mechanisms. *Neuropsychopharmacology*, 33(1), 18–41.
- [21] Lyle, C., Zheng, Z., Khetarpal, K., van Hasselt, H., Pascanu, R., Martens, J., & Dabney, W. (2024). Disentangling the causes of plasticity loss in neural networks. *arXiv preprint arXiv:2402.18762*.
- [22] Dohare, S., Hernandez-Garcia, J. F., Lan, Q., Rahman, P., Mahmood, A. R., & Sutton, R. S. (2024). Loss of plasticity in deep continual learning. *Nature*, 632(8026), 768–774.
- [23] Liu, J., Obando-Ceron, J., Courville, A., & Pan, L. (2024). Neuroplastic expansion in deep reinforcement learning. Published as a conference paper at ICLR 2025. *arXiv preprint arXiv:2410.07994v2*.
- [24] Uddin, L. Q. (2020). Bring the noise: Reconceptualizing spontaneous neural activity. *Trends in Cognitive Sciences*, 24(9), 734–746.
- [25] Northoff, G. (2018). *The Spontaneous Brain: From the Mind–Body to the World–Brain Problem*. MIT Press.
- [26] Ahmad, J. et al. (2022). From mechanisms to markers: novel noninvasive EEG proxy markers of the neural excitation and inhibition system in humans. *Translational Psychiatry*, 12(1), 467.
- [27] Swindale, N. V. et al. (2023). Spontaneous activity in cortical neurons is stereotyped and non-Poisson. *Cerebral Cortex*, 33(11), 6508–6525.
- [28] Northoff, G., Wainio-Theberge, S., & Evers, K. (2020). Is temporo-spatial dynamics the “common currency” of brain and mind? *Physics of Life Reviews*, 33, 34–54.
- [29] Northoff, G. & Zilio, F. (2022). Temporo-spatial Theory of Consciousness (TTC) – Bridging the gap of neuronal activity and phenomenal states. *Behavioural Brain Research*, 424, 113788.
- [30] 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, 4069. <https://doi.org/10.1038/s41467-020-17866-2>

- [31] Miconi, T., & Kay, K. (2025). Neural mechanisms of relational learning and fast knowledge reassembly in plastic neural networks. *Nature Neuroscience*, 28(2), 406–414. <https://doi.org/10.1038/s41593-024-01852-8>. PMID: 39814949.
- [32] Northoff, G. (2022). Temporo-spatial Theory of Consciousness—Bridging the gap between brain and consciousness. *Progress in Neurobiology*, 213, 102259.
- [33] Northoff, G., Wainio-Theberge, S., & Evers, K. (2019). Is temporo-spatial dynamics the “common currency” of brain and mind? *Physics of Life Reviews*, 31, 104–134.
- [34] Northoff, G. (2023a). As without, so within: How the brain’s temporo-spatial alignment shapes consciousness. *Interface Focus*, 13(3), 20220076.
- [35] Northoff, G. (2017). "Paradox of slow frequencies"—Are slow frequencies in upper cortical layers a neural predisposition of the level/state of consciousness? *Consciousness and Cognition*, 54, 20–35.
- [36] Huang, Z., Zhang, J., Wu, J., Mashour, G. A., & Northoff, G. (2023). Scale-free dynamics in core-periphery topography mediates consciousness. *Communications Biology*, 6(1), 499.
- [37] Canolty, R. T. & Knight, R. T. (2010). The functional role of cross-frequency coupling. *Trends in Cognitive Sciences*, 14(11), 506–515.
- [38] Hyafil, A., Giraud, A. L., Fontolan, L., & Gutkin, B. (2015). Neural cross-frequency coupling: connecting architectures, mechanisms, and functions. *Trends in Neurosciences*, 38(11), 725–740.
- [39] Northoff, G. (2023b). Temporospatial nestedness in consciousness. *Neuroscience Insights*, 18.
- [40] Raman, R., Kowalski, R., Achuthan, K., Iyer, A., & Nedungadi, P. (2025). Navigating artificial general intelligence development: societal, technological, ethical, and brain-inspired pathways. *Scientific Reports*, 15, 8443. <https://doi.org/10.1038/s41598-025-92190-7>
- [41] Li, J., Gu, B., & Huang, H. (2025). A fully single loop algorithm for bilevel optimization without hessian inverse. In *Proceedings of the AAAI Conference on Artificial Intelligence*. (As indicated by NSF PAR ID: 10533142, and seminar information regarding Junyi Li's work on nested ML and bilevel optimization in 2025).
- [42] Risto Miikkulainen (2025). Neuroevolution insights into biological neural computation. *Science*.
- [43] Wang, B., et al. (2025). Biologically inspired heterogeneous learning for accurate, efficient and low-latency neural network. *National Science Review*.
- [44] Sen, L., et al. (2022). Neuroevolution Guided Hybrid Spiking Neural Network Training. *Frontiers in Neuroscience*.
- [45] Faqiang , L., et al. (2024). Advancing brain-inspired computing with hybrid neural networks. *NLM*.
- [46] Lyle, C., et al. (2024). Disentangling the causes of plasticity loss in neural networks. *arXiv preprint arXiv:2402.18762*.
- [47] Zirui Huang (2023). Temporospatial Nestedness in Consciousness: An Updated Perspective on the Temporospatial Theory of Consciousness *Entropy*.
- [48] Hu, J., et al. (2024). High-Performance Temporal Reversible Spiking Neural Networks. *arXiv preprint arXiv:2405.16466*.
- [49] Omar Almomani (2021). A Hybrid Model Using Bio-Inspired Metaheuristic Algorithms for Network Intrusion Detection. *Computers, Materials and Continua*.
- [50] Mehonic1, A., et al. (2024). Roadmap to Neuromorphic Computing with Emerging Technologies *Nature Collection*.

- [51] Guerrero-Criollo, Roberto Jose, et al. (2023). Bio-inspired neural networks for decision-making mechanisms and neuromodulation for motor control in a differential robot. *Frontiers in Neurorobotics*.
- [52] Wess, C.; et al. *Natural Product-Inspired Dopamine Receptor Ligands*. *J. Med. Chem.* (2024), 67(15), 12712–12731. DOI: 10.1021/acs.jmedchem.4c00537.
- [53] Jacobsen, J. P. R.; et al. *The role of serotonin in depression—A historical roundup and future directions*. *J. Neurochem.* (2024), 168(10), 2113–2138. DOI: 10.1111/jnc.16097.
- [54] Carter, M. E.; et al. *Tonic and burst-like locus coeruleus stimulation distinctly shift brain network dynamics*. *Nat. Neurosci.* (2024), 27(11), 2168–2180. DOI: 10.1038/s41593-024-01755-8.
- [55] Likhtik, E.; Johansen, J. P. *Neuromodulator regulation and emotions: insights from the crosstalk of cell types in central amygdala*. *Mol. Psychiatry*. (2024), 29(5), 1312–1323. DOI: 10.1038/s41380-024-02415-0.
- [56] Hu, Y.; Zheng, Q.; Li, G.; Tang, H.; Pan, G. *Toward large-scale spiking neural networks: A comprehensive survey and future directions*. *arXiv preprint arXiv:2409.02111*, (2024).