

Article

Unsupervised Classification of Spike Patterns with the Loihi Neuromorphic Processor

Ryoga Matsuo , Ahmed Elgaradiny  and Federico Corradi * 

Department of Electrical Engineering, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands; ryoga.matsuo@epfl.ch (R.M.)

* Correspondence: f.corradi@tue.nl

Abstract: A long-standing research goal is to develop computing technologies that mimic the brain's capabilities by implementing computation in electronic systems directly inspired by its structure, function, and operational mechanisms, using low-power, spike-based neural networks. The Loihi neuromorphic processor provides a low-power, large-scale network of programmable silicon neurons for brain-inspired artificial intelligence applications. This paper exploits the Loihi processors and a theory-guided methodology to enable unsupervised learning of spike patterns. Our method ensures efficient and rapid selection of the network's hyperparameters, enabling the neuromorphic processor to generate attractor states through real-time unsupervised learning. Precisely, we follow a fast design process in which we fine-tune network parameters using mean-field theory. Moreover, we measure the network's learning ability regarding its error correction and pattern completion aptitude. Finally, we observe the dynamic energy consumption of the neuron cores for each millisecond of simulation equal to 23 $\mu\text{J}/\text{time step}$ during the learning and recall phase for four attractors composed of 512 excitatory neurons and 256 shared inhibitory neurons. This study showcases how large-scale, low-power digital neuromorphic processors can be quickly programmed to enable the autonomous generation of attractor states. These attractors are fundamental computational primitives that theoretical analysis and experimental evidence indicate as versatile and reusable components suitable for a wide range of cognitive tasks.

Keywords: neuromorphic computing; unsupervised learning; spiking neural networks; working memory; attractor dynamics



Citation: Matsuo, R.; Elgaradiny, A.; Corradi, F. Unsupervised

Classification of Spike Patterns with the Loihi Neuromorphic Processor.

Electronics **2024**, *13*, 3203. <https://doi.org/10.3390/electronics13163203>

Academic Editor: Shaogang Hu

Received: 9 June 2024

Revised: 18 July 2024

Accepted: 2 August 2024

Published: 13 August 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The concept of neuromorphic computing, as first presented in Mead's work [1], involves the creation of circuits, systems, and architectures that emulate the neuronal and synaptic dynamics of neurobiological systems. The main goal is to achieve computational efficiency and functionalities similar to those found in biological systems but within artificial devices.

One of the notable capabilities of the brain is decision-making, a vital function performed within specific cortical regions such as the parietal, prefrontal, and premotor areas [2,3]. Neurons within these regions display sustained activity during the brain's processing of sensory input, which it intends to retain within its working memory. Networks that exhibit self-sustained activity have the potential to maintain memory and facilitate the formation of bistable decisions, thus supporting various behaviors.

Importantly, slow-reverberating attractor networks offer a theoretical framework for understanding how decision-making circuits in the neocortex form categorical choices [4,5]. Bistability allows these networks to generate two distinct states with low and high firing rates, where the input stimulus level determines the specific state. This state representation is known as an attractor or associative memory. Such properties emerge based on a balanced interplay of excitatory and inhibitory interactions between neuronal populations [6,7].

Replicating this behavior within large-scale neuromorphic devices holds potential advantages in devising artificial intelligence systems with decision-making and working memory abilities that draw inspiration from the brain and are grounded in the same underlying physical phenomena. However, the challenge of implementing large-scale recurrent networks on neuromorphic devices through unsupervised online learning dynamics remains open. This is because training recurrent dynamics with traditional backpropagation methods requires storing the entire history of activations and the use of a surrogate gradient approach [8], making training a computationally expensive task that is not bio-realistic and subject to errors such as vanishing gradient and error accumulation. Many methods are being proposed to solve these issues. Recently, Yin et al., focused on addressing memory issues [9], while Deng et al., proposed a new surrogate model to reduce the accumulation of gradient errors [10]. However, these methods are computationally expensive and still require supervised training. Thus, this work explores neural parameters through a theory-guided methodology rather than training the spiking recurrent neural network via methods such as backpropagation or other supervised techniques.

In this work, we follow a theoretical approach proposed by Del Giudice et al. [11], which obviates the need for extensive preliminary parameter training of the network. Instead, the focus shifts towards configuring the network to be ready for learning. This approach allows the network to adapt the network's weights in an unsupervised fashion and avoids the need for external supervision.

We employ a mean-field theory approach and demonstrate the autonomous learning of attractor dynamics in a modern digital spike-based microchip, the Loihi neuromorphic processor [12]. This mean-field technique allows us to showcase on-chip distributed learning by creating self-sustained bi-stable activity (i.e., hi- and low-stable neuronal firing rates). Moreover, we demonstrate how to facilitate the system's rapid expansion into a more extensive network featuring multiple attractors, demonstrating the system's scalability.

Finally, we assess the on-chip system's error correction and pattern-completion abilities, and we evaluate the Loihi neuromorphic processor's power consumption as a benchmark for gauging the attractor network's performance.

In essence, the main contributions of this research are as follows:

- We exploit a mean-field theory-guided approach for unsupervised learning of attractor dynamics in the Loihi neuromorphic processor.
- We demonstrate the on-chip attractor network's pattern completion and error correction properties.
- We measure energy during unsupervised learning, infer attractor dynamics on the Loihi neuromorphic processor, and compare our results with analog and in-memory computing solutions.

2. Background

Over the past several decades, a significant body of research has been directed toward understanding how the brain's activity patterns correspond to the encoding of sensory inputs. These investigations aim to explain how sensory information is integrated and processed in the brain to generate contextual behavior. This includes but is not limited to, the translation of spike patterns into muscle movements [13], the formation and maintenance of working memories [14], spatial navigation [15], and also an abstract conceptual understanding as the concept of quantity, known as numerosity [16], or perceptual decision-making processes [5].

Within this context, attractor neural networks emerge as reusable components in all these functions. Attractor neural networks are recurrent neural networks in which the dimensionality of neural population activity is much lower than the population size. Attractor states refer to the stable patterns of neural activity that a network can converge to and maintain over time. These states are like "memories" that the network can recall when given an input similar to what it has learned before. In the context of this work, attractor states represent stable spiking frequencies of neuronal subpopulations that can persist

even in the absence of the initial stimulus. These states are crucial, as they form the basis for the network's ability to store and retrieve information, demonstrating capabilities like error correction and pattern completion. Moreover, the fact that recurrent neural networks in the brain exhibit attractor dynamics suggests a strategy that maps low-dimensional dynamics to a high-dimensional neural space as a standard feature of the brain's systems and is present in many functions, such as decision-making and working memories. While this method is but one tool in a more extensive arsenal, this strategy provides a link to the spiking neural activation, and it is used to explain and predict actions based on sensory perceptions. Additionally, the inherent match between attractor dynamics and the characteristics of biological neural systems, specifically their tolerance for noise, temporal imprecision, and hardware heterogeneity, facilitate integrating these dynamics into many neuromorphic platforms.

Self-sustained attractor dynamics and their formation can be modeled with integrate-and-fire neurons endowed with plastic synapses [11]. We employ a mean-field theory approach to simplify the analysis of recurrent networks composed of integrate-and-fire neurons. Mean field theory is a mathematical approach that is helpful in simplifying the analysis of large and complex networks by averaging the effects of all the individual components. In the context of our work, mean field theory helps us predict the network's behavior by focusing on the average input and output activities of neurons rather than tracking every single neuron's activity. This approach is instrumental in guiding the selection of network parameters, ensuring that the network can form stable attractor states. We can use mean-field theory to determine the effective transfer function and set parameters that facilitate the network's ability to exhibit attractor dynamics. This approach has often been used in computational neuroscience when modeling biological neural networks [3–5,11].

Additionally, these models have been mapped into ultra-low-power neuromorphic devices using a mixed-signal design style in several Very Large-Scale Integration (VLSI) technology nodes. These mixed-signal implementations have demonstrated ultra-low-power performance while emulating complex neuronal and synaptic dynamics [17,18]. In 2010, Camilleri et al. [19] investigated the use of mean-field theory to determine neuronal parameters and successfully designed an attractor network capable of exhibiting bistable spiking activity with low- and high-frequency firing states. The study demonstrated the behavior of the network implemented on silicon and its alignment with the effective transfer function (ETF) derived from the mean-field theory. Building upon this work, in 2015, Giulioni et al. in [20] demonstrated an attractor network in an analog VLSI system capable of learning to form two attractors and performing image classification of non-overlapping input stimuli from a silicon retina [21]. Other studies have also encoded attractor networks in mixed-signal chips and demonstrated bio-realistic behaviors, as the slow integration of perceptual evidence and the collapse in stable fixed points of network's dynamics [22–25] as modeled by computational neuroscientists [5]. More recently, in 2023, Cotteret et al. [26] demonstrated stable persistent-firing attractor dynamics in an analog on-chip network consisting of a hard Winner-Take-All SNN by implementing silicon neurons with excitatory recurrent synapses. Although analog neurons better capture the behavior of real biological neurons [27], the size of these networks has been constrained by the engineering and fabrication effort required. For these reasons, Field Programmable Gate Array (FPGA) implementations have also been proposed to model attractor dynamics [28–31]; these allow the programming of many biophysically realistic synaptic models with slow and non-linear synapses, which are essential in emulating slow ramp-up activities in perceptual decision-making tasks.

In order to advance beyond these endeavors, our investigation focuses on the applicability of such theories in a modern, large-scale digital neuromorphic device, the Intel Loihi. The Loihi chip offers the advantage of being a large-scale spiking neural network device realized with an advanced 14 nm technology node and can simulate up to 128 thousand neurons within an area of 0.41 mm². The newer version of Intel, the Loihi-2, is fabricated in

a 4 nm technology node and can host up to 1 million neurons within an area of 0.21 mm², i.e., smaller than a fingertip. This quick scaling in large network sizes has challenges, such as longer programming times. Therefore, it is crucial to develop programming methods that enable rapid and efficient configuration of these devices without extensive pre-training. The ease of programming digital neuromorphic devices is critical to ensuring their widespread adoption and practical use of large-scale neuromorphic devices.

3. Methods

3.1. The Loihi Neuromorphic Processor

The Loihi neuromorphic processor [12] is an Intel research platform. It is a digital, barrier-synchronized design for large-scale programmable spiking neural networks. The chip architecture has 128 neuromorphic cores, each functioning independently of the others. Each core processes 1024 neural units sequentially, which can be organized into dendritic tree structures to shape neurons. Unlike other digital neuromorphic platforms (e.g., SpiNNaker [32–34] and SENECA [35,36]), Loihi's default is to send barrier messages between cores, ensuring system-wide synchronization. This approach offers the benefit of a consistent and predictable system response but may lag due to the slowest core. However, compared to a system that is synchronized by a global clock, this method ensures that each time step is as brief as necessary. Additionally, the cores are linked by a two-level 2D mesh network that operates both within the chip (intra-chip) and between chips (inter-chip). This routing mesh facilitates direct spike communication between cores and manages synchronizing tasks. A spike can be sent sequentially to a set of destination cores and then disseminated to a specific group of neurons within each core. The digital distribution of these spikes also incorporates adaptable connectivity templates to facilitate standard neuron pool connections, such as fully connected and convolutional filters.

Additionally, Loihi offers diverse features that interest researchers who explore the creation of large-scale SNNs. It allows for adding stochastic noise to its internal states, the integration of discrete delays for spikes and postsynaptic currents, and the capability for synapses to exhibit complex dynamics. Further, dendritic trees can be formed by repurposing neural units in each core. Neuron models can incorporate adaptive components; learning rules are framed using a universal blueprint (see Section 3.7 for more details) and support multiple weight compression methods, spike broadcasting, and integrated control through built-in x86 cores. Furthermore, in the latest version of Loihi, the Loihi 2 device, features such as graded spikes [37] and the efficient implementation of convolutional synaptic connectivity [38] have been introduced.

Being completely digital and synchronized through barriers, Loihi offers fewer obstacles to users than mixed-signal, solely analog, or entirely asynchronous digital solutions. This ensures that the chip operates dependably and consistently. However, there are still challenges when contrasting Loihi to the simulation of SNNs on conventional hardware, such as Intel x86 CPUs or NVIDIA GPUs. These include quantizing or truncating actual values due to a limited bit precision and constraints on connectivity and storage. Nevertheless, these effects can be considered thanks to the open-source Lava software framework (developed by Intel Corporation (Santa Clara, CA, USA) and available via <https://github.com/lava-nc/lava> (accessed on 8 June 2024)). The framework facilitates the efficient placement of spiking neural networks (SNNs) on Loihi, optimizing power consumption and accuracy while considering hardware constraints.

3.2. Neuron Model

The neuron model exploits the compartment parameters set via the NxSDK-2.0 software package [39] (a Python package provided by Intel Corp.). NxSDK adopts the leaky integrate and fire model based on current (CUBA) that has two internal states: the synaptic response current $u_i(t)$ and the membrane voltage $v_i(t)$. This model integrates the spikes into the incoming current and accumulates them as the membrane voltage. The current and voltage decay with configurable time constants in a low-pass filter fashion. As the

membrane voltage exceeds the neuron's firing threshold, the neuron emits a spike, and the voltage is reset to 0 (i.e., the reset potential H is set to $H = 0$). The synaptic response current is the sum of filtered input spike trains and a constant bias current as defined in [12] and expressed as:

$$u_i(t) = \sum_{j \neq i} w_{i,j} (\alpha_u \cdot \sigma_j)(t) + b_i, \quad (1)$$

where $w_{i,j}$ is the synaptic efficacy from presynaptic neuron j to postsynaptic neuron i , $\sigma(t) = \sum_k \delta(t - t_k)$ is the spike train represented by Dirac functions where t_k is the k_{th} spike timing, $\alpha_u(t) = \tau_u^{-1} \exp(-\frac{t}{\tau_u}) H(t)$ is the synaptic filter impulse response parameterized by the current decay time constant τ_u with the $H(t)$ unit step function, and b_i is a constant bias. This current is further integrated into steps to derive the membrane voltage, and neurons emit a spike as the voltage exceeds the firing threshold θ_i . Hence, the membrane voltage dynamics can be expressed as

$$\dot{v}_i(t) = -\frac{1}{\tau_v} v_i(t) + u_i(t) - \theta_i \sigma_i(t), \quad (2)$$

where τ_v is the voltage decay time constant [12]. The first term ($-\frac{1}{\tau_v} v_i(t)$) represents the membrane voltage decay β parameterized by the time constant, the second term ($u_i(t)$) represents the incoming synaptic response current to the neuron, and the third term ($-\theta_i \sigma_i(t)$) represents the reset spikes generated when the voltage exceeds the firing threshold. We must note that we use the terms 'voltage' and 'currents' to describe the membrane potential and its internal dynamics, aligning our model with the behavior of actual biological neurons. However, in the context of the hardware implementation at hand, these terms are purely symbolic and represent digital values, not literal electrical properties, and thus, we omit physical units. The compartment parameters for our objectives are the firing threshold θ , the voltage decay time constant τ_v , the current decay time constant τ_u and the refractory period τ_{ref} . The refractory period is the time period in which the neuron does not integrate the incoming spikes to recover the initial state from the previous spike event and is not represented in the above equations. The firing threshold θ and the current decay time constant τ_u are set to $\theta = 11,520$ and $\tau_u = 1$ time step, respectively, using the NxSDK-2.0 API. The voltage decay time constant τ_v and the refractory period τ_{ref} are tuned using the procedures introduced in Section 3.4 and set to $\tau_v = 16$ time step and $\tau_{ref} = 3$ time step.

3.3. Design of Attractor Network

The neuronal network configuration used to establish the attractor dynamics on the Loihi neuromorphic chip is detailed in Figure 1. Spike inputs are produced through on-chip spike generators denoted as S_{in} , which introduce input stimuli with spike timing following a Poisson distribution to each neuron within the excitatory population denoted as E (implying individual connectivity). For the inhibitory stimulator, every pair of spike generators S_{in} delivers an input stimulus to each neuron within the inhibitory population labeled I , representing a two-to-one connectivity pattern, as inhibitory neurons are half of the excitatory ones.

During the training phase, the synaptic strengths originating from the spike generators to the excitatory population denoted $J_{E,S_{in}}$, and to the inhibitory population, denoted $J_{I,S_{in}}$, are set at $J_{E,S_{in}} = 0.194$ and $J_{I,S_{in}} = 0.167$, respectively. These efficacy values are normalized with the neuron's firing threshold value ($\theta = 11,520$) and represent the instantaneous increase (or decrease) in the membrane potential upon the arrival of a presynaptic spike. Furthermore, E_{noise} and I_{noise} introduce background noise into each neuron within the excitatory and inhibitory populations, maintaining a one-to-one connectivity pattern. Specifically, E_{noise} and I_{noise} are configured to emit spikes at 10 spikes per 100 time steps and 50 spikes per 100 time steps, respectively. These noise-induced spiking rates are used throughout the rest of this work and emulate background noise from distant neural populations. The inhibitory population I makes only inhibitory connections, including

self-recurrent ones, to balance excitation and inhibition. Moreover, this design aligns with Dale’s principle, which states that neurons release only one type of neurotransmitter and have the same functional effect at all termination sites [40].

The synaptic strengths associated with the noise generators within the excitatory population, labeled $J_{E,E_{noise}}$, and within the inhibitory population, labeled $J_{I,I_{noise}}$, are established at $J_{E,E_{noise}} = J_{I,I_{noise}} = 0.056$.

Within this architecture, the count of excitatory neurons is represented as $128P$, while the number of inhibitory neurons stands as $64P$, where P denotes the capacity of the network to grasp distinct spike patterns (i.e., the number of E, I pairs). In the context of this paper, the terminology “subpopulation” designates each group of 128 excitatory neurons, and each of these subpopulations is tasked with acquiring the ability to construct an attractor.

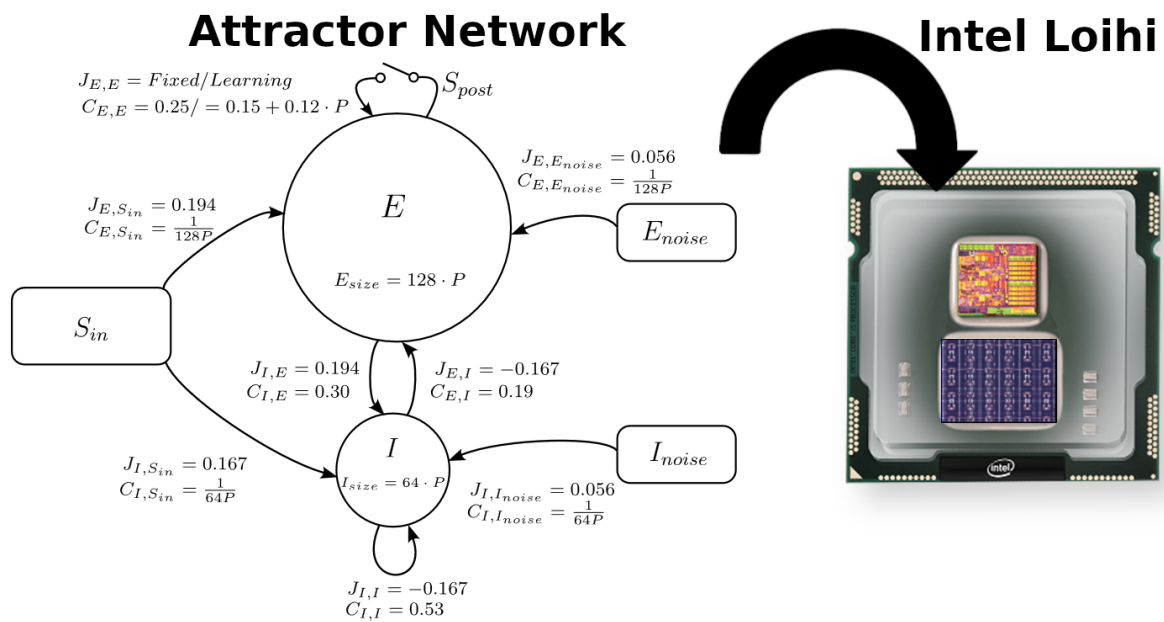


Figure 1. Template network architecture programmed into the Loihi device. The switch at the excitatory recurrent connections is open for the measurement of the ETF and closed for training and inference. The scaling factor, P , corresponds to how many spike patterns the network can learn (i.e., each group of 128 excitatory neurons learns one spike pattern). Note that some values of efficacies are changed in the learning and inference phase to encourage activity for faster learning at low recurrent efficacies of the excitatory population. For those adjusted efficacies, the values used for learning and inference are indicated on the left and right sides, respectively.

The connections between neural populations are drawn from a flat random distribution and are as follows: from excitatory to excitatory neurons, represented as $C_{E,E}$ (constituting recurrent connections within the excitatory population), the connectivity equals 0.25. This means that each neuron in the excitatory population has a probability of 0.25 to connect to any other neuron in the target population (excitatory). Furthermore, the excitatory to inhibitory connections indicated as $C_{I,E}$ stand at 0.30, and the inhibitory to excitatory connections indicated as $C_{E,I}$, at 0.19. Connectivity within the inhibitory population, $C_{I,I}$, indicating the recurrent connections of inhibitory neurons, is established at 0.53.

The effectiveness of connections originating from excitatory neurons to inhibitory neurons and indicated by $J_{I,E}$, is set at a value of $J_{I,E} = 0.194$. On the contrary, the strengths of connections from inhibitory to excitatory ($J_{E,I}$) and from inhibitory to inhibitory ($J_{I,I}$) populations are both set at $J_{E,I} = J_{I,I} = -0.167$.

Finally, the recurrent strengths within the excitatory population, represented by $J_{E,E}$, are learned from spiking activity to facilitate the construction of attractors. The learning rule for these connections is addressed in Section 3.7.

3.4. Theory-Guided Parameter Tuning

Neuronal parameters must be thoroughly calibrated to devise an attractor network that can exhibit self-sustained bistable activity. These parameters are determined by evaluating the effective transfer function (ETF) while considering the spiking frequencies of recurrent presynaptic and postsynaptic neurons and subsequently pinpointing the network stability states through the self-consistency equation.

The mean-field self-consistency equation posits that a network of interconnected neurons exhibits (meta) stability at frequency points where the mean input and output frequencies are equal. The mean output firing frequency is expressed as $\nu_{out} = \Phi(\mu, \sigma)$, where Φ is the population transfer function, and μ and σ indicate the mean and variance of the presynaptic current, respectively. The stability line is represented by the solid black line in Figure 2 and is expressed as $\nu_{out} = \nu_{in}$. In this context, ν_{in} and ν_{out} denote presynaptic and postsynaptic spiking frequencies, respectively.

The mean postsynaptic frequency of a neuronal population is expressed as [41]

$$\begin{aligned} \nu_{out} &= \Phi(\mu, \sigma) \\ &= \left[\tau_{ref} + \frac{\sigma^2}{2\mu^2} \left(e^{\frac{-2\mu(\theta-H)}{\sigma^2}} - 1 + \frac{2\mu(\theta-H)}{\sigma^2} \right) \right]^{-1}, \end{aligned} \quad (3)$$

where τ_{ref} , θ , and H represent the refractory period, firing threshold, and reset potential of the neurons in the network. By measuring this non-linear transfer function, we can find the range of excitatory recurrent efficacies $J_{E,E}$ that satisfies the self-consistency equation and thus the bistability. ETF measurements are obtained by first disconnecting excitatory recurrent connections by opening the switch shown in Figure 1 and injecting presynaptic spike inputs S_{pre} with different spike frequencies into the excitatory population. We sweep the mean presynaptic spiking frequency from 1 Hz to 350 Hz, and we measure the mean output frequency of the postsynaptic neurons of the excitatory population. Connectivity $C_{E,S_{pre}}$ is set to 0.25 (that is, the same as $C_{E,E}$). Figure 2 shows the measured ETFs for four different efficacies: $J_{E,S_{pre}} = 0.028, 0.083, 0.117$ and 0.122 . For recurrent efficacy $J_{E,E} = 0.028$, there is only one intersection at 0.0 Hz, meaning the system converges to the resting state for any input stimulus. For recurrent efficacy $J_{E,E} = 0.117$, the ETF intersects the stability line at 0 spikes per 100 time steps, at slightly less than 2 spikes per 100 time steps, and around 12 spikes per 100 time steps. For recurrent efficacy equal to $J_{E,E} = 0.122$, intersections are found at 0 spikes per 100 time steps, slightly above 2 spikes per 100 time steps, and at about 24 spikes per 100 time steps. When the ETF intersects three times the stability line, the first and third points are stable, and the second point is metastable for both of these efficacies. Hence, when the network is stimulated by some input and then released from the stimulus, any level of recurrent spiking activity that is below the metastable point converges to the lowest stable state of activity, and any level of recurrent spiking activity that is above the metastable point converges to the high-frequency stable state. From these measurements, the recurrent synaptic efficacies must evolve to an average value above $J_{E,E} = 0.117$ to endow the network with a metastable and highest stable state (investigated in Section 4). As a result of this analysis, the compartment parameters that affect the shape of the ETF, the membrane voltage decay, and the refractory period of neurons are chosen to be $\tau_v = 16$ time steps and $\tau_{ref} = 3$ time steps, respectively. These values are determined by measuring the ETFs for different parameter values and selecting a combination that satisfies the bistability.

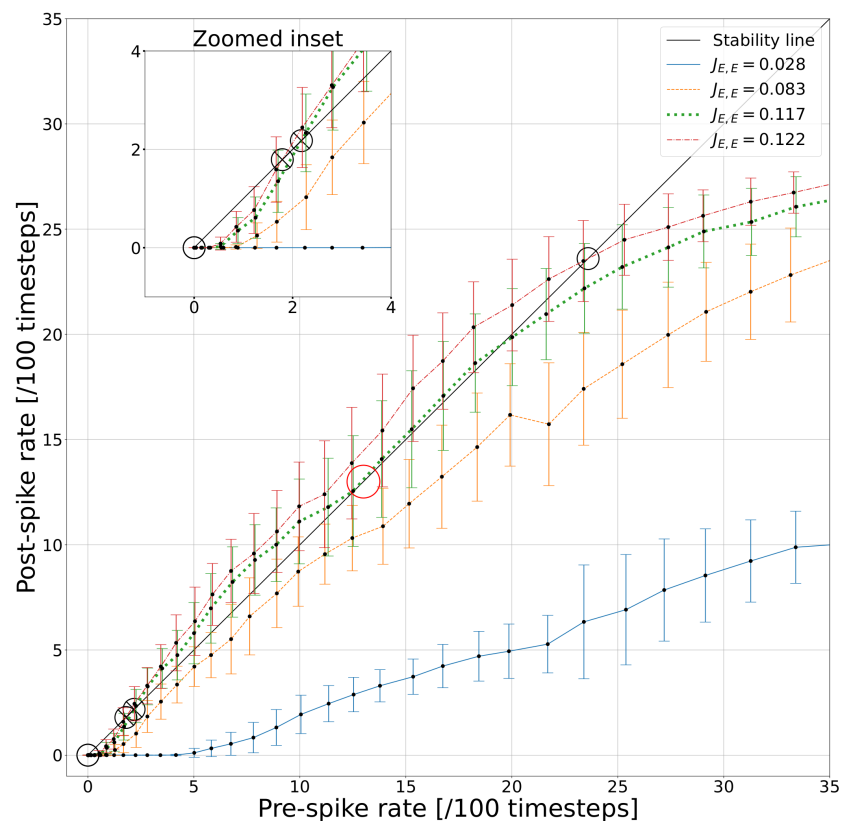


Figure 2. Effective transfer function. The three lines represent the input (x-axis) and output (y-axis) average firing frequencies of the excitatory population in the function of three distinct recurrent synaptic weights. The error bars represent the standard deviation of the mean frequency. The mean efficacy values as indicated in the inset are $\langle J_{E,E} \rangle (S_{pre}) = 0.028$ (solid line), $\langle J_{E,E} \rangle (S_{pre}) = 0.083$ (dashed line), $\langle J_{E,E} \rangle (S_{pre}) = 0.117$ (dotted line), and $\langle J_{E,E} \rangle (S_{pre}) = 0.122$ (dash-dotted line). Empty circles “○” represent stable points of the network dynamics, while crossed circles “⊗” indicate meta-stable points of the network dynamics.

3.5. Mapping Attract Networks in the Loihi Neuromorphic Processor

When mapping a neural network onto neuromorphic hardware, one must consider the limitations of the available hardware resources because of the direct mapping of neuron and synaptic models and parameters into physically limited resources. In constructing connectivity, Loihi imposes the following restrictions on the mapping and design of the attractor network [12]:

1. The maximum number of neurons in any given core must not exceed 1024.
2. The maximum fan-in state mapped to any given core must not exceed 128 KB.
3. The maximum number of core-to-core fan-out connections stored on the output side of any given core must not exceed 4096.
4. The maximum number of *axon_id* stored on the input side of any given core must not exceed 4096.

Given the extensive synaptic connections of the network, neurons are spread over multiple cores to meet the final three constraints. As a result, each subgroup consisting of 128 excitatory neurons and 64 inhibitory neurons is allocated to an individual core. In total, P cores are used to house the entire neuron population as depicted in Figure 3a. This method reduces hardware utilization efficiency, as there are 192 neurons in use out of 1024 neurons available per core. However, this is not the primary emphasis of this study.

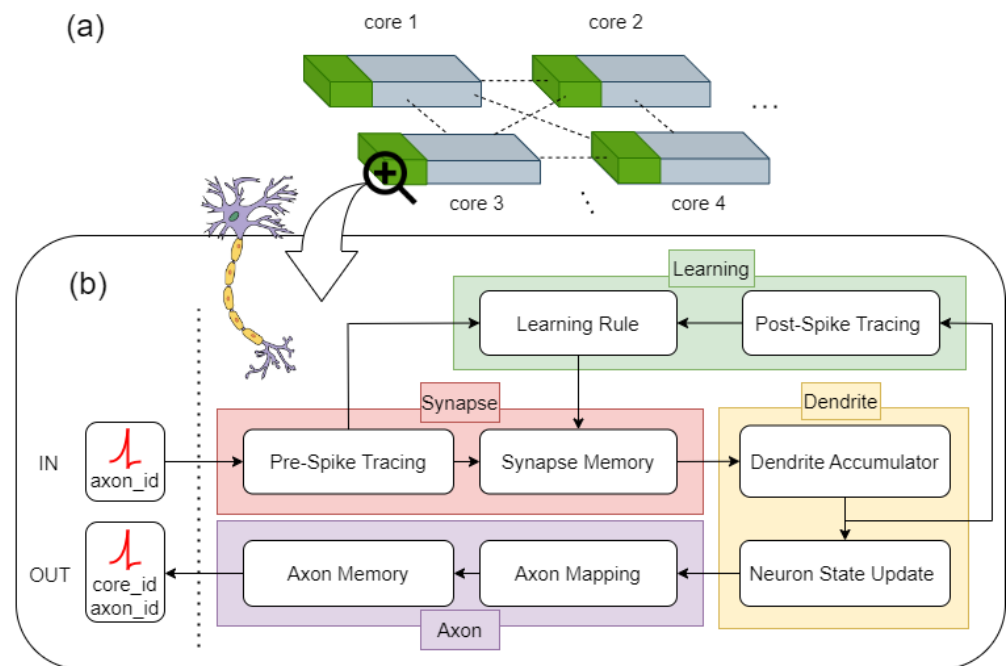


Figure 3. Mapping neurons to the Loihi neuromorphic processor. (a) The processor contains multiple cores, each depicted as a grey block. Each core can support a maximum of 1024 neurons. The attractor network is distributed across these cores, with the utilized sections shown in green. Within each utilized section, there are 192 neurons, with a division of 128 excitatory neurons and 64 inhibitory neurons. (b) A streamlined flowchart illustrates how the neurons and synapses in the Loihi processor operate.

Each neuron consists of multiple functional blocks at the lower level to mimic biological neuronal mechanisms, as depicted in Figure 3b. As a spike enters the neuron, the *axon_id* of the presynaptic neuron becomes stored to be used in learning the synaptic efficacy. Then, the synaptic efficacy is fetched and fed into the dendrite accumulator. The state of the neuron is updated according to the accumulated change. If the voltage threshold is reached and the neuron fires, the *axon_id* and the *core_id* of the postsynaptic neurons are fetched, and the message is transmitted through the axon. This hardware model is simplified to facilitate the understanding. In reality, more mechanisms such as parallelism and read–modify–write memory accesses are deployed at the hardware level to accelerate and ensure the correct computation.

3.6. Attractor Network Dynamics

Upon fine-tuning the parameters in accordance with the ETF and the self-consistency equation, the recurrent connections are made operational, and the number of neuronal subpopulations is set to 1. This step is undertaken to validate the potential formation of an attractor at the stable points. To achieve this, we establish the recurrent excitatory strengths at $J_{E,E} = 0.122$. The outcome is illustrated in Figure 4.

In the setup, spike generators S_{in} are configured to generate presynaptic spikes, approximately 15 spikes per 100 time steps from 0 to 500, about 33 spikes per 100 time steps from 500 to 1000, and again around 15 spikes per 100 time steps from 1000 to 1500. The initial and final inputs symbolize subtle stimuli, while the intermediate input signifies a robust stimulus.

As depicted in Figure 4, the neural spiking activity remains suppressed during the initial stimulus and escalates during the final stimulus despite both being subjected to the same stimulation frequency. This phenomenon suggests the network experiences stimulation near its metastable point when introducing a weak stimulus. Slight deviations from this point prompt the network to transition to either a low- or high-activity state. In

this instance, the low-frequency state is initially triggered beneath the metastable point, followed by the strong stimulus inducing the high-activity state of attractor dynamics. Remarkably, this elevated state persists even after the cessation of the high-frequency input, indicating working memory dynamics.

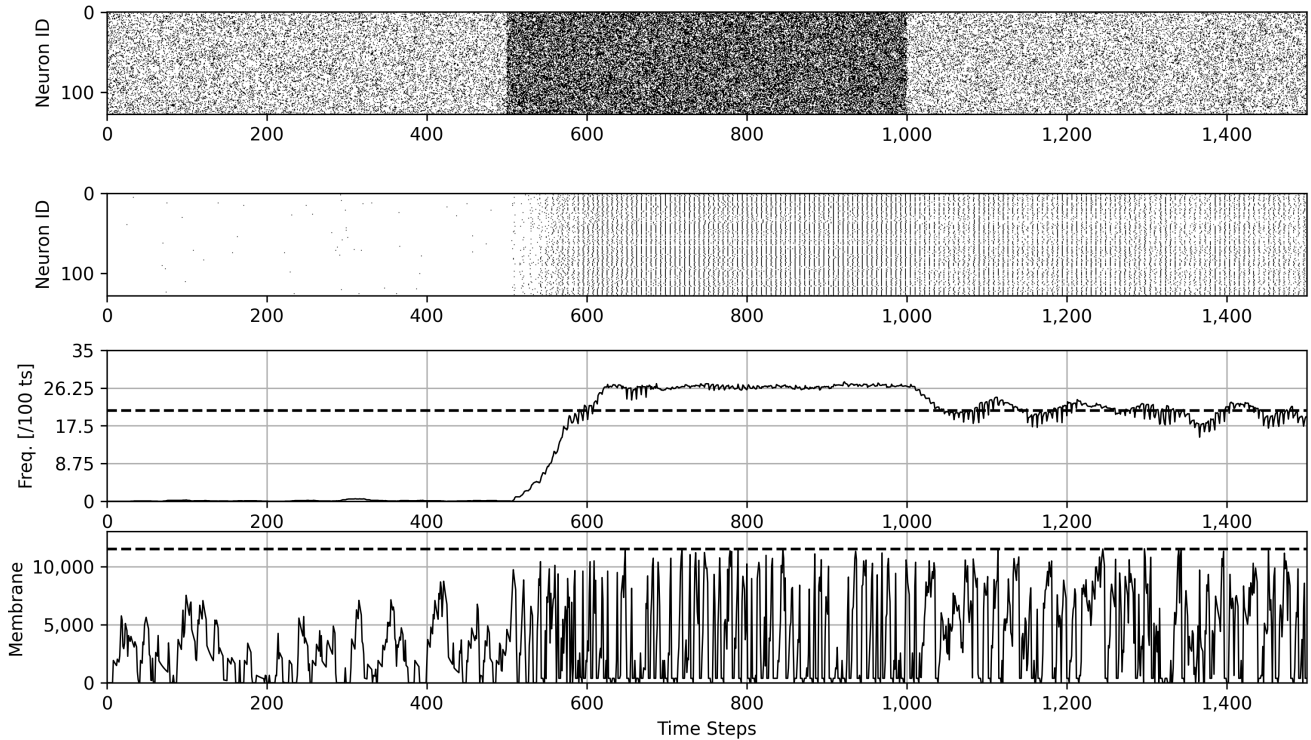


Figure 4. Attractor dynamics for one neuronal population, i.e., $P = 1$, and the excitatory recurrent efficacies set at $J_{E,E} = 0.122$. **(First Row)** Raster plot of presynaptic spikes of the excitatory neurons, **(Second Row)** raster plot of postsynaptic spikes of the excitatory neurons, **(Third Row)** peristimulus time histogram (PSTH) representing the mean spiking frequencies of the excitatory neurons. The dotted line indicates the stable point of the network dynamics as measured from the ETF. **(Fourth Row)** membrane voltage in one of the excitatory neurons.

Figure 4 demonstrates the stability around 24 spikes per 100 time steps, which is in agreement with the ETF measurement for the recurrent excitatory efficacy of $J_{E,E} = 0.122$ (as measured in Section 3.4).

3.7. Learning Rule

Once the recurrent efficacy to form an attractor is determined and confirmed, the learning rule is formulated so that the attractors are learned through autonomous online learning. We introduce a spike-timing dependent plasticity (STDP) learning rule [42], in which the efficacy of each recurrent connection $J_{E,E}$ is updated every two time steps. This rule can mathematically be represented as

$$\Delta J_{E,E} = (2^{-3}x_1y_0 - 2^{-3}y_1x_0 - \text{sgn}(J_{E,E} - 0.139) \cdot 2^{-4}x_1y_0 - 2^{-4}x_1y_0), \quad (4)$$

where x_0 , y_0 , x_1 , and y_1 represent the presynaptic spike, postsynaptic spike, presynaptic spike trace, and postsynaptic spike trace, respectively. This equation is derived from the programmable learning rule already available in the Loihi as described in [39]. Moreover, the spikes can be modeled as digital quantities, and the traces are real values. The presynaptic trace x_1 is an impulse generated at the occurrence of the presynaptic spike x_0 , and it decays exponentially over time. This trace is multiplied by the postsynaptic spike y_0 to track the degree of correlation of the spike time between the presynaptic and

postsynaptic terminals. Similarly, the postsynaptic trace y_1 is an impulse generated at the occurrence of the postsynaptic spike y_0 , and it decays exponentially over time. This trace is multiplied by the presynaptic spike x_0 to obtain the degree of decorrelation in the timing of the spike between the presynaptic and postsynaptic terminals. Both of these traces, x_1 and y_1 , are characterized by the initial impulse of 20 and the decay time constant of 4 to learn the dynamics of the attractor as can be seen in Equation (4); the time correlation increases synaptic efficacy, and the time decorrelation decreases synaptic efficacy. The last two terms regulate the synaptic efficacy from increasing to a value greater than $J_{E,E} = 0.139$. They cancel each other out when the efficacy is lower than $J_{E,E} = 0.139$, resulting in no effect. When the efficacy is greater than $J_{E,E} = 0.139$, the sum of the two terms cancels out the first term, and the efficacy can stay constant or decrease depending on the degree of decorrelation. From Figure 2, it is evident that $J_{E,E} = 0.117$ yields attractor stability and performance compared to $J_{E,E} = 0.122$. However, any value above 0.117 provides a stable fixed point due to the nature of recurrent connectivity. We use a threshold of $J_{E,E} = 0.139$, as determined by the ETF exploration, to regulate the synaptic efficacy. This threshold ensures that the synaptic efficacy does not exceed 0.139, maintaining network stability and attractor dynamics.

Since this formulation of the learning rule does not require disabling the learning after convergence, the learning is on the fly, and it automatically enables the stopping of learning without the use of any external signal. Such balancing mechanisms enable us to leave on-the-fly learning always on, as once stability is reached (with a value greater than 0.139), there is no further effect. Maximum efficacy is selected to be $J_{E,E} = 0.139$, as most of the synaptic efficacy remains at the initial value as derived in Section 4.1. We note that the purpose of training is to evolve the average excitatory recurrent efficacy of a pool of excitatory neurons to a value greater than $J_{E,E} = 0.117$ as determined in Section 3.4 so that the network can exhibit attractor dynamics.

3.8. Experimental Setup and Configuration of the Loihi Neuromorphic Processor

The Loihi neuromorphic processor features 128 neuromorphic cores, each capable of simulating up to 1024 neurons. The network topology is initialized according to the design illustrated in Figure 1, encompassing multiple groups of excitatory and inhibitory neurons arranged to form attractor networks. Initial excitatory recurrent strengths $J_{E,E}(0)$ are set to a small value to facilitate the gradual formation of attractor states. Additionally, the membrane voltage decay time constant (τ_v) and the refractory period (τ_{ref}) are tuned to 16 and 3 time steps, respectively, to ensure appropriate neuronal dynamics. Multiple groups of 128 spike generators are configured to excite the neurons at different, non-overlapping time intervals, creating distinct sets of spatial activations within the excitatory population. Background noise is introduced to both excitatory and inhibitory populations using noise generators configured to emit spikes at 10 spikes per time step and 50 spikes per 100 time steps, respectively, to emulate the natural background noise found in biological neural systems. The synaptic strengths from spike generators to the excitatory ($J_{E,Sin}$) and inhibitory ($J_{I,Sin}$) populations are set at 0.194 and 0.167, respectively. Synaptic strengths associated with noise generators ($J_{E,Enoise}$ and $J_{I,Inoise}$) are set to 0.056 for both excitatory and inhibitory populations. Recurrent connectivity within the excitatory population ($C_{E,E}$) and between excitatory and inhibitory populations ($C_{I,E}$, $C_{E,I}$, $C_{I,I}$) is initialized with a connectivity probability of 0.25 for excitatory and 0.52 for inhibitory populations. Mean field theory is employed to experimentally measure the effective transfer function (ETF) using mean input currents from a pool of interconnected neurons. This approach helps predict stable activity points and identify parameter ranges that ensure stability and attractor dynamics. Furthermore, the spike-timing-dependent plasticity (STDP) learning rule is implemented to adjust synaptic efficacies based on the timing of pre- and postsynaptic spikes, allowing for on-the-fly fine-tuning of the network and reducing the need for extensive pretraining and manual adjustments.

For running the simulations, the network is stimulated with spike trains from the configured spike generators, each set to activate neuron groups at specified intervals. After each stimulation, a strong spike train is sent to the inhibitory population to reset the excitatory population's activity, ensuring it is returned to the low-frequency state. The activity of neuronal subpopulations is monitored over time to observe the formation of self-sustained attractor states, tracking the spiking frequencies and membrane voltages of neurons. The network's effectiveness in forming stable attractors, as well as its error correction and pattern completion properties, is assessed by providing partial and ambiguous stimuli and measuring the accuracy of attractor retrieval. Dynamic energy consumption during the learning and inference phases is measured using the Nahuku32 board, which houses 32 Loihi chips. This involves monitoring the power usage and computing the energy consumption per time step. In order to support these analyses, we use several software libraries including matplotlib for plotting, nxsdk for interfacing with the Loihi hardware, and numpy and pandas for data manipulation and analysis. We also make use of utilities from the nxsdk package, such as plotRaster and graph monitoring probes, to visualize neural activity and efficiently manage simulation data.

3.9. Attractor Formation via Unsupervised Stimuli

The methodology for forming stable attractors involves initializing the network topology as shown in Figure 1 with the initial excitatory recurrent strength $J_{E,E}(0)$ set to a small value. Spike generators then excite multiple groups of neurons at different times, with non-overlapping time intervals. Specifically, to utilize the network for a classification task, P groups of 128 spike generators, $S_{in,t1}-S_{in,tP}$ are created (see Figure 5). Each group is allocated a different time interval at which the generators in the respective group emit spikes. In other words, the groups of generators inject non-overlapping/orthogonal sets of spike patterns at given time intervals. In addition, after every stimulation by a group of spike generators, strong resetting spikes are injected into the inhibitory population to reset the membrane activity of the excitatory population and prevent the previous stimulus from impacting the result of the current stimulus. Furthermore, the input stimulus is switched from one group to another in multiple iterations such that the synapses in different neuronal subpopulations learn the spike patterns as uniformly as possible at any arbitrary time step. Importantly, attractors are formed by altering the recurrent connectivity strengths. This is achieved thanks to a spike-timing-dependent plasticity (STDP) learning rule that adjusts the synaptic efficacies based on the timing of pre- and postsynaptic spikes (see Section 3.7). Over time, the activity of the neuronal subpopulations increases, forming self-sustained attractors. These attractors exhibit stable spiking frequencies even in the absence of the stimulus.

These attractors act as distributed memory and demonstrate robust error correction and pattern completion properties. These properties are demonstrated in Section 4.2 by providing partial stimulation and observing full memory retrieval and by providing ambiguous stimulation in which several groups of neurons are stimulated with some overlap. The extent of partial stimulation (overlap) is measured using the Hamming distance, which represents the number of differences between two sets of inputs.

To evaluate our attractor network's error correction and pattern completion capabilities implemented on the Loihi neuromorphic processor, we conduct a series of simulations using a custom-built spiking neural network. The network is configured with multiple excitatory and inhibitory neuron populations, and the synaptic connections are established using pre-trained weight matrices. To simulate erroneous inputs, we generate target neurons with varying portions of induced errors (bit flips). Spike generators are employed to create precise spike timings for stimuli, noise, and reset signals, ensuring distinct activation patterns across the network. The network's activity is monitored during each simulation by probing spikes and membrane voltages. The mean frequency of output spikes is calculated for each subpopulation over time, allowing us to assess the stability and performance of the attractor states.

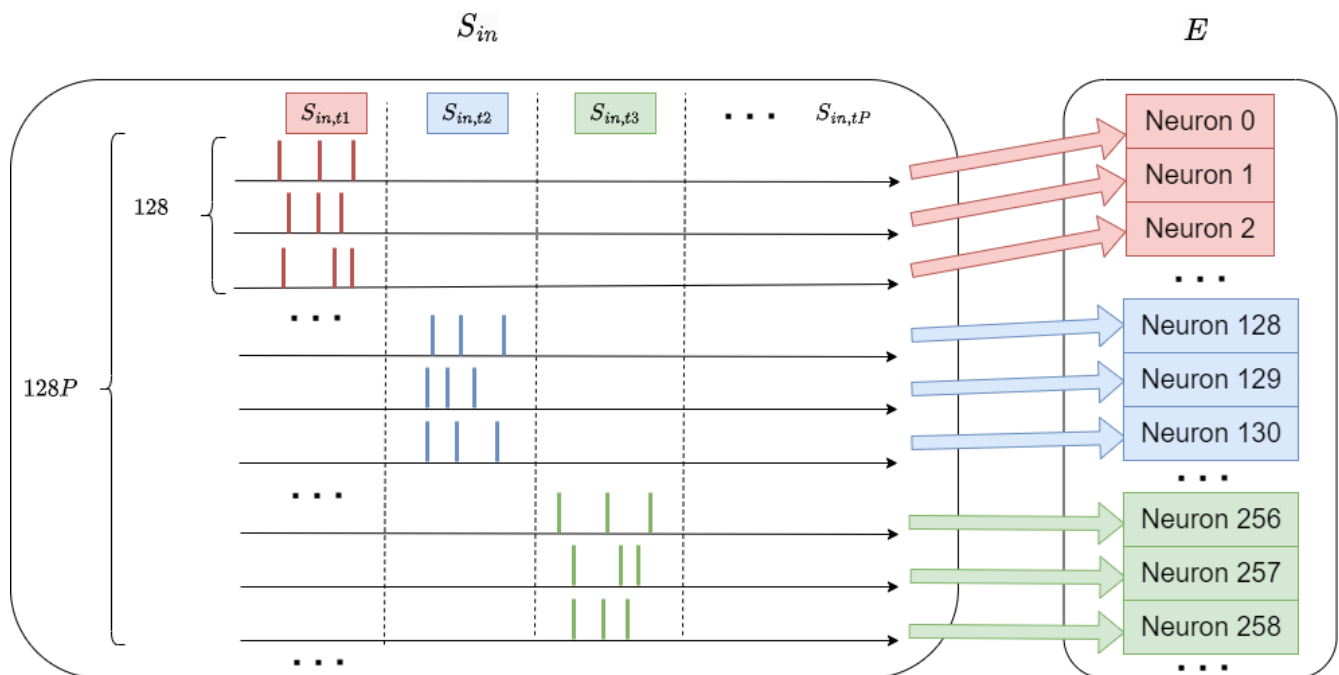


Figure 5. Spike generators, S_{in} injecting orthogonal/non-overlapping spike patterns into the excitatory neuronal population, E . The dashed vertical lines indicate the time of different stimuli onsets.

To quantify error correction, we measure the network's ability to retrieve the original attractor states despite introduced errors. An attractor is considered successfully retrieved if the mean frequency of output spikes during the resting intervals exceeds a specified threshold (e.g., 10 spikes per 100 time steps). Mathematically, if $F_{\text{retr}}(t)$ represents the mean frequency of the retrieved attractor at time t , the attractor is deemed successful if $F_{\text{retr}}(t) \geq 10$ spikes per 100 time steps during the evaluation period. Pattern completion is evaluated by providing partial stimuli and observing the network's capacity to generate the complete pattern from incomplete inputs. The results are analyzed by plotting raster plots of pre- and postsynaptic spikes, membrane voltages, and spiking activity over time, which provide a comprehensive understanding of the network's dynamic behavior and robustness in handling noisy and partial inputs. In Section 4.2, we provide quantitative analysis and experimental results.

4. Results

4.1. Unsupervised Learning of Attractor Dynamics

Here, we provide evidence of the network's ability to autonomously learn the attractor dynamics without requiring external supervision. We initialize the initial excitatory recurrent strengths $J_{E,E}(0)$ to zero, and we employ continuous online learning.

The results for four input groups ($P = 4$) are shown in Figure 6. It can be observed that the activity of the four subpopulations of neurons increases over time as the learning progresses. These subpopulations start showing self-sustained activities, hence forming attractors around 25,000 time steps. Furthermore, in the magnified peristimulus time histogram (PSTH) plot, the spiking frequencies of the four attractors are sustained at around 15 spikes per 100 time steps. This result matches the stability analysis as predicted using the ETF measurements.

Furthermore, the synaptic matrix of the recurrent connections is plotted in four different time instances in Figure 7a. An increase in the efficacies contained in squares at the diagonal positions can be observed with time progression. The synaptic efficacy distributions in these squares (i.e., recurrent connections within each subpopulation) are plotted over time in Figure 7b. As can be seen, most of the synapses stayed at low efficacies, except for a few that learn up to high efficacies. These results indicate that learning occurs in the

subset of neurons that fire together and that the network achieves a high degree of sparsity. Table 1 reports average recurrent efficacies in the four neuronal subpopulations. It can be observed that the synapses learned quickly until their efficacies converge to values in the range of $0.125 < J_{E,E} < 0.130$, which is consistent with the theoretical value determined in Section 3.4. Using the recorded synaptic matrices from the Loihi device, we can determine the evolution of the effective transfer function for the four subpopulations. Figure 8 shows the ETF for the four subpopulations at three time instances during the learning. At time step 7650, the ETF shows the only stable point exists around 0 frequency (intersect between diagonal and ETF for all subpopulations), while at time 15,300, the subpopulations 2 and 3 start reaching the diagonal line, indicating attractor formation. Finally, at time step 30,600, all subpopulations reach the diagonal line and intersect it in three points: a low-frequency state, an unstable point, and a stable point at high frequency.

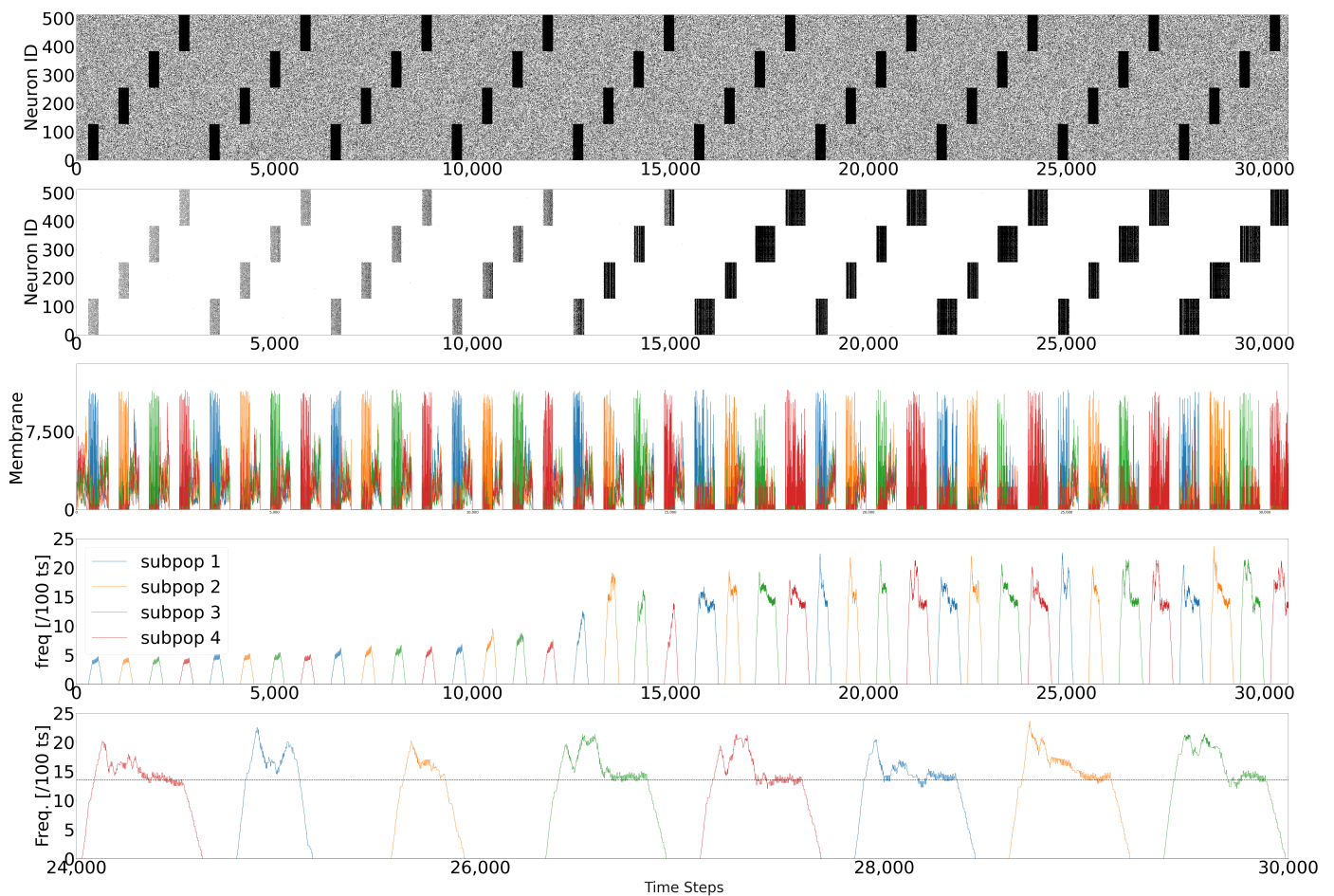


Figure 6. Unsupervised learning of attractor dynamics. **(First row)** Raster plot of presynaptic spikes of the excitatory neurons, **(Second row)** raster plot of postsynaptic spikes of the excitatory neurons, **(Third row)** membrane voltage in one of the excitatory neurons within each subpopulation, **(Fourth row)** PSTH within each subpopulation, and the **(Fifth row)** shows magnified plot of the **(Fourth row)**. The black dotted line indicates the stable point of the subpopulations as measured from the ETF.

Table 1. Average recurrent efficacies of four neuronal subpopulations over time.

Time Step	$J_{E,E}^{Subpop1}$	$J_{E,E}^{Subpop2}$	$J_{E,E}^{Subpop3}$	$J_{E,E}^{Subpop4}$
0	0	0	0	0
7650	0.017	0.030	0.015	0.014
15,300	0.048	0.126	0.095	0.044
22,950	0.127	0.121	0.112	0.109
30,600	0.125	0.127	0.119	0.130

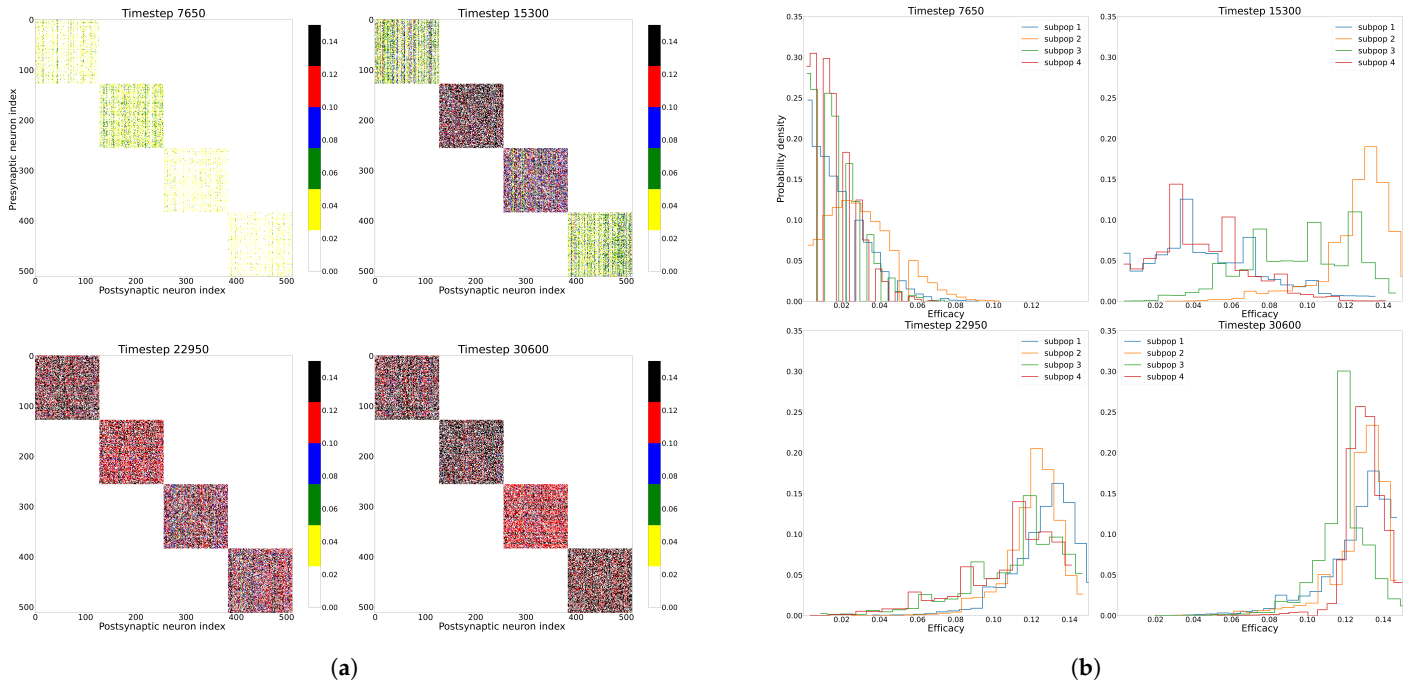


Figure 7. Synaptic evolution during training. **(a)** The synaptic matrix of the excitatory recurrent connections at different time steps t during training. The color of each dot indicates the efficacy of a specific synapse connecting the presynaptic and postsynaptic neurons. The synaptic matrix at $t = 0$ is completely blank, as all synaptic efficacies are set at 0. We note that only 25% of the excitatory neurons are recurrently connected (i.e., $C_{E,E} = 0.25$). **(b)** The probability distributions of excitatory recurrent efficacies within four different subpopulations at different time steps t . In this scenario, each colored square in **(a)** is a subpopulation.

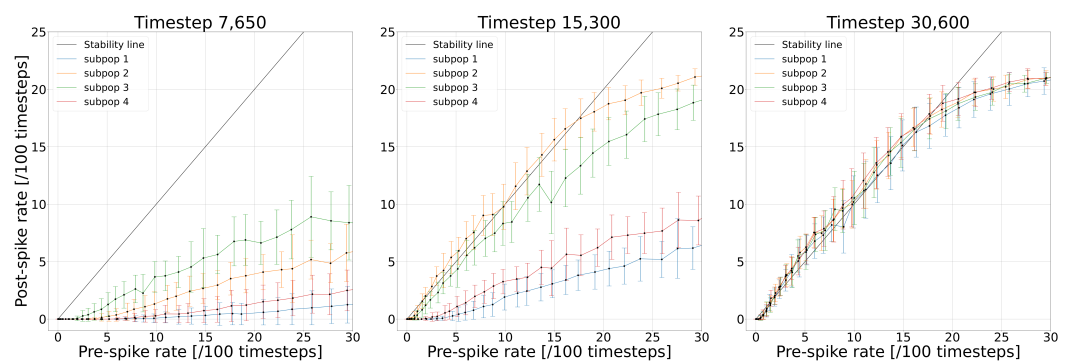


Figure 8. Evolution of the effective transfer function of the excitatory population during the learning process.

4.2. Error Correction and Pattern Completion

The attractors' error correction and pattern completion capabilities are demonstrated using the synaptic strengths learned from the attractor dynamics. To probe the error-correction properties of attractors, we exchange the stimuli with neurons in other attractors (i.e., we perform cross-stimulation of multiple attractors). In this context, the count of altered stimuli corresponds to the Hamming distance, where each alteration equates to a bit flip. The reconstruction accuracies of the four attractors and the average accuracy across them are reported in Figure 9a. The threshold where the accurate reconstruction starts to decline (a reconstruction accuracy of 50%) emerges at approximately a Hamming distance of 90–100.

This range of values is expected, as when four attractors are trained in an entirely uniform manner (that is, their synaptic strengths are uniform), successful attractor reconstruction requires stimuli from $\frac{1}{4} \cdot 100 = 25\%$ of neurons within the respective attractor to overcome the impact of stimuli from other attractors. Thus, to correctly reconstruct an attractor, 32 out of 128 neurons must be stimulated concurrently (yielding a Hamming distance of 96, which aligns with the expected range).

In our network, a gap between the reconstruction accuracy of attractor 3 and the other attractors is apparent. This discrepancy arises because the synaptic strengths are not learned in a perfectly uniform manner due to the stochastic nature of the attractor network. This observation is consistent with the fact that the average efficacy of recurrent connections in subpopulation 3 (at time step 30,600) is the highest among all neuronal subpopulations, as demonstrated in Table 1.

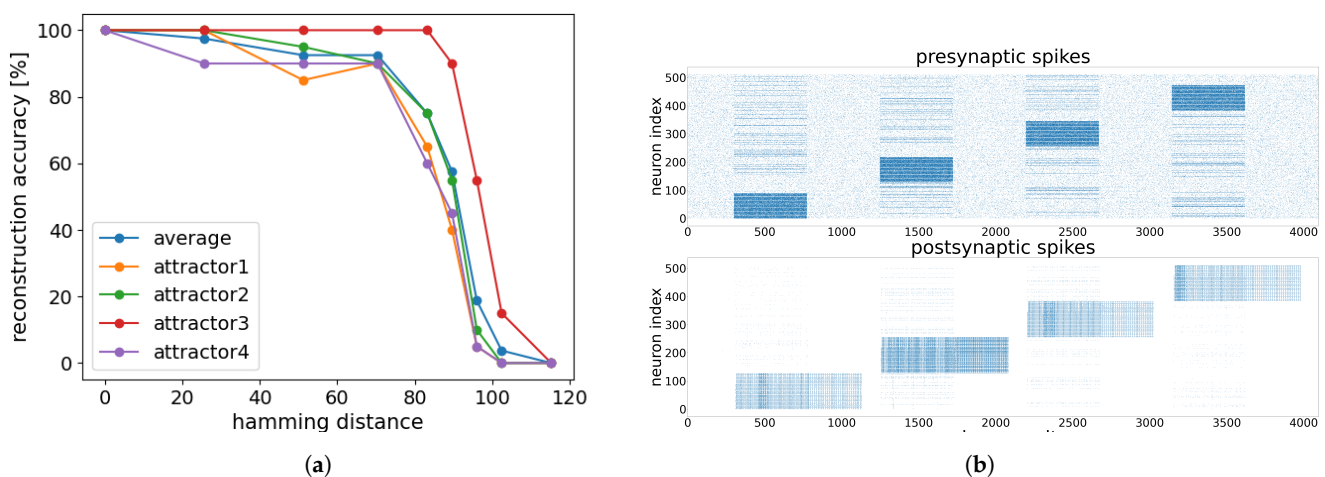


Figure 9. (a) Accuracies of attractor retrieval. (b) Ambiguous stimulation with a Hamming distance of 30. Attractors are successfully retrieved.

4.3. Energy Profiling

The power consumption for the attractor network comprising four attractors, each composed of 512 excitatory neurons and 256 inhibitory neurons, is carried out using the Nahuku32 board, which houses 32 Loihi chips.

The summarized data, including the average power measurements in five trials and the computed energy consumption per time step for the learning and inference tasks, are shown in Table 2. It is important to note that only Neurocores (i.e., cores housing neurons within Loihi) [39] are utilized in these programs. Consequently, the power consumption of CPU cores responsible for time and power analysis is not factored into the energy calculation. Furthermore, the static power of Neurocores, which encompasses power leakage from both logic and memory, is omitted from consideration, as it does not directly correlate with computation during execution.

Table 2. Performance profiling on Nahuku32.

	Learning	Inference
Execution time per time step (μs)	3974	4658
Total power (mW)	786	808
Total static power (mW)	768	789
Total dynamic power (mW)	18	19
Static power in Neurocores (mW)	4	5
Dynamic power in Neurocores (mW)	6	5
Dynamic energy in Neurocores ($\mu\text{J}/\text{time step}$)	23	22

Hence, the dynamic energy consumption per time step within Neurocores during the learning phase is determined to be 23 $\mu\text{J}/\text{time step}$, while during the inference phase, it amounts to 22 $\mu\text{J}/\text{time step}$.

Dynamic power consumption is lesser than static power consumption because there is no continuous need to transfer data, especially when adjusting efficacies in the inference phase. On the other hand, static power is higher because it involves keeping a large amount of data, notably the efficacies, stored in memory.

5. Discussion

Traditional methods for tuning neural network parameters, such as backpropagation, often rely on supervised training techniques. These methods require large amounts of data and numerous iterations to find optimal parameters, making them computationally intensive and time-consuming due to the need to store and process large volumes of activation histories and gradients. In contrast, our approach leverages experimental and theoretical methods, specifically mean field theory, to guide the selection and tuning of network parameters and does not require the calculation of any gradient. This makes our method particularly compatible with physical systems, such as neuromorphic hardware and analog circuits, where gradient calculations could be more complex and practical. In these systems, the continuous real-time nature of signal processing and the constraints of physical components make traditional gradient-based methods infeasible. By eliminating the need for gradient calculations, our method allows for efficient parameter tuning and adaptation directly within the hardware, facilitating more practical and scalable implementations of neural networks in such environments. The process we propose in this paper involves three main steps; first, we conduct an experimental analysis using mean field behaviors to measure the effective transfer function (ETF), which predicts stable activity points and identifies parameter ranges that ensure stability and attractor dynamics. Second, by focusing on average behaviors predicted by the ETF, we preselect a narrower range of parameters likely to yield stable and effective network performance. And third, we initialize the network in a state where only the efficacy of recurrent connections needs adjustment to reach stable attractor states. We employ spike-timing-dependent plasticity (STDP) for on-the-fly fine-tuning synaptic efficacies based on spike timing. This approach offers several advantages: it is efficient, providing a high-level overview of network dynamics and reducing trial-and-error iterations; it reduces computational demands by avoiding extensive activation history storage and gradient calculations; and it accelerates the learning process through real-time adaptation, eliminating the need for extensive supervised training datasets.

Additionally, we adopt a method in which attractor dynamics function in a population-based style without overlap or with minimal overlap, ensuring that one attractor does not unintentionally activate another. However, this significantly limits the number of memorizable attractor states. Potential solutions could be found in integrating more bio-realistic neural mechanisms, such as spike frequency adaptations and homeostatic plasticity [43], that can enhance the network's stability. By incorporating these mechanisms,

it could have been feasible for the network to acquire information from spike patterns without altering the efficacy of input synapses as the system transitions from the training to the testing phase. This adaptation ensures that neurons maintain appropriate spiking frequencies, facilitating ongoing learning.

Due to constraints related to computational resources necessary for processing and storing measurement data on the server, the network's runtime poses challenges. The prolonged operation, exceeding 1,250,000 time steps, is demanding in terms of memory storage for spiking activity, and observing a gradual progression in synaptic plasticity becomes challenging. These limitations also constrain the scale of the model, resulting in a smaller configuration featuring only four subpopulations, each comprising 768 neurons.

The stochastic nature of the attractor network leads to variations in synaptic efficacy in different subpopulations. This, in turn, results in discrepancies in learning and attractor activation as observed in the reconstruction accuracy of different attractors. Fine-tuning learning parameters, such as learning rate and timing, could mitigate this issue and foster more uniform synaptic strengths.

Achieving a balance between synaptic plasticity and network stability emerges as a central challenge. While synaptic strengths are needed to evolve for learning, they also require stability to maintain attractor dynamics. The complex interplay between these factors underscores the need for dynamic mechanisms that adjust synaptic weights while preventing excessive interference between subpopulations.

Finally, even without these improvement possibilities, our experiment shows the extensive potential of attractor neural networks. The theory-guided approach enhances the scalability of the network within state-of-the-art large-scale neuromorphic systems such as the Intel Loihi.

5.1. Attractor Networks in Neuromorphic Hardware: Comparison and Trade-Offs

Even if what constitutes a synaptic operation differs from one neuromorphic architecture to another, neuromorphic systems that operate on spikes exploit the energy per synaptic operation (E_{SOP}) metric to assess energy consumption. This metric captures the energy consumed in both synaptic activities and spike generation. Table 3 summarizes the results for E_{SOP} for five neuromorphic processors in which spike-based attractor dynamics are implemented.

From Table 3, we note that mixed-signal neuromorphic devices can result in greater energy efficiency than their digital counterparts; for example, the Braindrop mixed-signal device is about four times more efficient than the Tianjic chip at peak performance. Importantly, attractor dynamics appear to be well suited for analog neuromorphic systems, as they can accommodate mismatch, noise, and other deviations. This is guaranteed when realizing attractor dynamics with sufficiently large networks in which currents and rates are estimated using average net contributions, thus facilitating analog computation.

Table 3. Comparison of energy per synaptic operation E_{SOP} in seven recent neuromorphic processors. E_{SOP} are obtained by dividing the total chip power consumption by the SOP rate. For the Tianjic chip, the E_{SOP} is derived from the peak performance of 650 giga synaptic operation per Watt reported in [44]. All measures come from the reference listed in the first row.

	Loihi [12]	SpiNNaker [45,46]	SpiNNaker2 [47]	Tianjic [48,49]	Braindrop [50]	DynapSE [51,52]	BSS-2 [53]
Technology	14 nm FinFet	130 nm	22 nm FDSOI	28 nm HLP	28 nm FDSOI	180 nm	65 nm LP
Supply voltage (V)	0.50 V–1.25 V	1.2 V	0.45–0.6	0.85 V	1 V	1.3 V	0.9 V–1.26 V
Design Style	digital	digital	digital	digital	mixed-signal	mixed-signal	mixed-signal
E_{SOP}	>23.6 pJ	>26.6 nJ	10–20 pJ	1.54 pJ	381 fJ	2.8–17 pJ	O (10 pJ)

Attractor dynamics are compatible with noisy but energy-efficient neuromorphic platforms because neural and synaptic imperfections and inhomogeneities are averaged

when estimating the stable points of the dynamics; this is because we exploit average net input and output currents. As a result, spike-based attractor neural networks are appealing for emerging, ultra-low-power computing materials that, while energy-efficient, are affected by noise and inconsistencies. Persin et al. in [54] and Wang et al. in [55] demonstrate, in simulations, that the unsupervised learning method can be applied in memristor-based devices, thanks to the robust use of attractor dynamics.

5.2. Practical Implications and Potential Integrations

Neuromorphic computing contrasts artificial neural networks (ANNs) based on back-propagation, which has been extensively studied and implemented on modern computer systems by using biology-inspired algorithms such as spiking neural networks (SNNs) endowed with bio-inspired learning mechanisms like spike-timing-dependent plasticity (STDP), which are still in their early stages of research and development. While ANNs have achieved significant success and dominate the field due to advanced graphics processing units and vast training datasets, they are hindered by the conventional von Neumann computing architecture. This architecture leads to high energy consumption during both training and inference operations because of massive and frequent data movements, posing substantial power efficiency and performance challenges. In contrast, non-volatile-memory-based processing-in-memory neuromorphic computing architectures offer a promising alternative, enabling neural networks to operate with much higher parallelism and reduced data transfer [56]. However, these non-volatile-memory-based systems suffer from noise, endurance, and variability issues.

Thus, the findings from our research demonstrate advancements in error correction and pattern completion capabilities within neuromorphic computing, highlighting the potential for integrating attractor networks into existing and emerging technologies. Our approach, which leverages STDP for real-time learning, is particularly well suited for applications requiring continuous adaptation and robust performance in the presence of noise and errors. This makes it ideal for integrating memory technologies based on emerging nanotechnologies [57,58], which often exhibit unreliable behavior and require constant learning and adaptation. By incorporating attractor network methodologies, these systems can achieve greater stability and reliability, enhancing their overall functionality.

The properties of real-time processing and low-energy requirements make neuromorphic technologies a perfect candidate for robotic systems integration. In these systems, it is possible to integrate neural-inspired computational primitives thanks to neuromorphic processors like Intel's Loihi, which offer substantial energy efficiency and benefits from real-time adaptation. In particular, the use of spiking neural networks (SNNs) to mimic brain computation methods demonstrates remarkable energy efficiency, consuming significantly less power than traditional von Neumann architectures for a variety of tasks [12], which is crucial for practical robotics applications. Moreover, the incorporation of spike-timing-dependent plasticity (STDP) and decision-making behaviors using the same computational primitives of the human brain, i.e., attractor dynamics, could enable real-time learning and dynamic adaptation to new tasks and environments, enhancing versatility and reliability in autonomous navigation and interaction. Attractor dynamics has already been shown in the Loihi device for exploiting the inherent dynamics of spiking neuron models to solve problems by converging to well-defined fixed points in phase space [59]. Our work demonstrates the on-the-fly learning of attractor dynamics and their robustness through enhanced error correction and pattern completion, which is essential for high-precision tasks like object recognition and manipulation in unstructured environments. The scalability and flexibility of neuromorphic hardware like Loihi allow the rapid reconfiguration of robotic systems without extensive pretraining, supporting advanced cognitive functions and improving decision-making capabilities essential for complex tasks.

6. Conclusions

We experimentally demonstrate the unsupervised formation of attractor states in the Loihi neuromorphic processor. To streamline the network design process, we utilize the effective transfer function (ETF) and the self-consistency equation to fine-tune the compartment parameters. Learning the dynamics of the attractor is possible due to the online STDP learning rule, which enables unsupervised on-chip learning. The trained network achieves accurate retrieval of spike patterns with extensive (~50%) corruption, demonstrating its error correction and pattern completion capabilities. Finally, the attractor network achieves a dynamic energy consumption of a 23 μJ / per time step. These results highlight the potential of neuromorphic computing systems, which can provide robust, scalable, and energy-efficient solutions by enabling the use of emerging nanomaterials in future neuromorphic computing technologies.

Author Contributions: F.C. conceived the experiments. R.M. and F.C. wrote the experiment routines (in Python code). R.M., A.E. and F.C. conducted the experiments. F.C. analyzed the results. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The code for reproducing the experiment can be obtained at this link (GitHub) <https://github.com/federicohyo/attractordynamicsintel/> (accessed on 8 June 2024).

Acknowledgments: We express our sincere gratitude to the Intel Neuromorphic Research Community for their invaluable assistance and generous provision of hardware resources. Their technical support has contributed significantly to successfully executing our research endeavors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Mead, C. *Analog VLSI Implementation of Neural Systems; The Kluwer International Series in Engineering and Computer Science*; Springer: Boston, MA, USA, 1989; Volume 80. [CrossRef]
2. Mattia, M.; Pani, P.; Mirabella, G.; Costa, S.; Del Giudice, P.; Ferraina, S. Heterogeneous attractor cell assemblies for motor planning in premotor cortex. *J. Neurosci.* **2013**, *33*, 11155–11168. [CrossRef] [PubMed]
3. Deco, G.; Rolls, E.T. Attention, short-term memory, and action selection: A unifying theory. *Prog. Neurobiol.* **2005**, *76*, 236–256. [CrossRef] [PubMed]
4. Rolls, E.T. *Memory, Attention, and Decision-Making: A Unifying Computational Neuroscience Approach*; Oxford University Press: Oxford, UK, 2007.
5. Wang, X.J. Probabilistic Decision Making by Slow Reverberation in Cortical Circuits. *Neuron* **2002**, *36*, 955–968. [CrossRef]
6. Gigante, G.; Mattia, M.; Braun, J.; Del Giudice, P. Bistable perception modeled as competing stochastic integrations at two levels. *PLoS Comput. Biol.* **2009**, *5*, e1000430. [CrossRef] [PubMed]
7. Brinkman, B.A.; Yan, H.; Maffei, A.; Park, I.M.; Fontanini, A.; Wang, J.; La Camera, G. Metastable dynamics of neural circuits and networks. *Appl. Phys. Rev.* **2022**, *9*, 011313. [CrossRef]
8. Zhang, W.; Li, P. Spike-train level backpropagation for training deep recurrent spiking neural networks. *Adv. Neural Inf. Process. Syst.* **2019**, *32*. Available online: https://proceedings.neurips.cc/paper_files/paper/2019 (accessed on 8 June 2024).
9. Yin, B.; Corradi, F.; Bohté, S.M. Accurate online training of dynamical spiking neural networks through Forward Propagation Through Time. *Nat. Mach. Intell.* **2023**, *5*, 518–527. [CrossRef]
10. Deng, S.; Lin, H.; Li, Y.; Gu, S. Surrogate Module Learning: Reduce the Gradient Error Accumulation in Training Spiking Neural Networks. In Proceedings of the 40th International Conference on Machine Learning, Honolulu, HI, USA, 23–29 July 2023.
11. Del Giudice, P.; Fusi, S.; Mattia, M. Modelling the formation of working memory with networks of integrate-and-fire neurons connected by plastic synapses. *J. Physiol. Paris* **2003**, *97*, 659–681. [CrossRef] [PubMed]
12. Davies, M.; Srinivasa, N.; Lin, T.H.; Chinya, G.; Cao, Y.; Choday, S.H.; Dimou, G.; Joshi, P.; Imam, N.; Jain, S.; et al. Loihi: A neuromorphic manycore processor with on-chip learning. *IEEE Micro* **2018**, *38*, 82–99. [CrossRef]
13. Gallego, J.A.; Perich, M.G.; Miller, L.E.; Solla, S.A. Neural manifolds for the control of movement. *Neuron* **2017**, *94*, 978–984. [CrossRef]
14. Axmacher, N.; Mormann, F.; Fernández, G.; Cohen, M.X.; Elger, C.E.; Fell, J. Sustained neural activity patterns during working memory in the human medial temporal lobe. *J. Neurosci.* **2007**, *27*, 7807–7816. [CrossRef]
15. Miller, J.F.; Neufang, M.; Solway, A.; Brandt, A.; Trippel, M.; Mader, I.; Hefft, S.; Merkow, M.; Polyn, S.M.; Jacobs, J.; et al. Neural activity in human hippocampal formation reveals the spatial context of retrieved memories. *Science* **2013**, *342*, 1111–1114. [CrossRef] [PubMed]

16. Viswanathan, P.; Nieder, A. Neuronal correlates of a visual “sense of number” in primate parietal and prefrontal cortices. *Proc. Natl. Acad. Sci. USA* **2013**, *110*, 11187–11192. [[CrossRef](#)]
17. Chicca, E.; Stefanini, F.; Bartolozzi, C.; Indiveri, G. Neuromorphic electronic circuits for building autonomous cognitive systems. *Proc. IEEE* **2014**, *102*, 1367–1388. [[CrossRef](#)]
18. Indiveri, G.; Linares-Barranco, B.; Hamilton, T.J.; Schaik, A.V.; Etienne-Cummings, R.; Delbruck, T.; Liu, S.C.; Dudek, P.; Häfliger, P.; Renaud, S.; et al. Neuromorphic silicon neuron circuits. *Front. Neurosci.* **2011**, *5*, 73. [[CrossRef](#)]
19. Camilleri, P.; Giulioni, M.; Mattia, M.; Braun, J.; Del Giudice, P. Self-sustained activity in attractor networks using neuromorphic VLSI. In Proceedings of the International Joint Conference on Neural Networks, Barcelona, Spain, 18–23 July 2010. [[CrossRef](#)]
20. Giulioni, M.; Corradi, F.; Dante, V.; Del Giudice, P. Real time unsupervised learning of visual stimuli in neuromorphic VLSI systems. *Sci. Rep.* **2015**, *5*, 14730. [[CrossRef](#)] [[PubMed](#)]
21. Lichtsteiner, P.; Posch, C.; Delbruck, T. A 128×128 120 dB $15 \mu\text{s}$ latency asynchronous temporal contrast vision sensor. *IEEE J. Solid-State Circuits* **2008**, *43*, 566–576. [[CrossRef](#)]
22. Pfeil, T.; Grübl, A.; Jeltsch, S.; Müller, E.; Müller, P.; Petrovici, M.A.; Schmuker, M.; Brüderle, D.; Schemmel, J.; Meier, K. Six networks on a universal neuromorphic computing substrate. *Front. Neurosci.* **2013**, *7*, 11. [[CrossRef](#)]
23. Qiao, N.; Mostafa, H.; Corradi, F.; Osswald, M.; Stefanini, F.; Sumislawska, D.; Indiveri, G. A reconfigurable on-line learning spiking neuromorphic processor comprising 256 neurons and 128K synapses. *Front. Neurosci.* **2015**, *9*, 141. [[CrossRef](#)]
24. Corradi, F.; You, H.; Giulioni, M.; Indiveri, G. Decision making and perceptual bistability in spike-based neuromorphic VLSI systems. In Proceedings of the 2015 IEEE International Symposium on Circuits and Systems (ISCAS), Lisbon, Portugal, 24–27 May 2015; pp. 2708–2711. [[CrossRef](#)]
25. Partzsch, J.; Mayr, C.; Giulioni, M.; Noack, M.; Hänzsche, S.; Scholze, S.; Höppner, S.; Giudice, P.D.; Schüffny, R. Mean field approach for configuring population dynamics on a biohybrid neuromorphic system. *J. Signal Process. Syst.* **2020**, *92*, 1303–1321. [[CrossRef](#)]
26. Cotteret, M.; Richter, O.; Mastella, M.; Greatorex, H.; Janotte, E.; Girão, W.S.; Ziegler, M.; Chicca, E. Robust Spiking Attractor Networks with a Hard Winner-Take-All Neuron Circuit. In Proceedings of the 2023 IEEE International Symposium on Circuits and Systems (ISCAS), Monterey, CA, USA, 21–25 May 2023; pp. 1–5.
27. Zendrikov, D.; Solinas, S.; Indiveri, G. Brain-inspired methods for achieving robust computation in heterogeneous mixed-signal neuromorphic processing systems. *Neuromorphic Comput. Eng.* **2023**, *3*, 034002. [[CrossRef](#)]
28. de Vangel, B.C.; Torres-Huitzil, C.; Girau, B. Spiking dynamic neural fields architectures on fpga. In Proceedings of the 2014 International Conference on ReConfigurable Computing and FPGAs (ReConFig14), Cancun, Mexico, 8–10 December 2014; pp. 1–6.
29. Chappet De Vangel, B.; Torres-Huitzil, C.; Girau, B. Randomly spiking dynamic neural fields. *ACM J. Emerg. Technol. Comput. Syst.* **2015**, *11*, 1–26. [[CrossRef](#)]
30. de Vangel, B.C.; Torres-Huitzil, C.; Girau, B. Event based visual attention with dynamic neural field on FPGA. In Proceedings of the 10th International Conference on Distributed Smart Camera, Paris, France, 12–15 September 2016; pp. 142–147.
31. You, H.; Zhao, K. Neuromorphic Implementation of a Continuous Attractor Neural Network with Various Synaptic Dynamics. *IEEE Access* **2021**, *9*, 109224–109240. [[CrossRef](#)]
32. Furber, S.B.; Galluppi, F.; Temple, S.; Plana, L.A. The spinnaker project. *Proc. IEEE* **2014**, *102*, 652–665. [[CrossRef](#)]
33. Furber, S.; Bogdan, P. *Spinnaker—A Spiking Neural Network Architecture*; Now Publishers: Boston, MA, USA; Delft, The Netherlands, 2020. [[CrossRef](#)]
34. Yan, Y.; Stewart, T.C.; Choo, X.; Vogginger, B.; Partzsch, J.; Höppner, S.; Kelber, F.; Eliasmith, C.; Furber, S.; Mayr, C. Comparing Loihi with a SpiNNaker 2 prototype on low-latency keyword spotting and adaptive robotic control. *Neuromorph. Comput. Eng.* **2021**, *1*, 014002. [[CrossRef](#)]
35. Yousefzadeh, A.; Van Schaik, G.J.; Tahghighi, M.; Detterer, P.; Traferro, S.; Hijdra, M.; Stuijt, J.; Corradi, F.; Sifalakis, M.; Konijnenburg, M. SENECA: Scalable energy-efficient neuromorphic computer architecture. In Proceedings of the 2022 IEEE 4th International Conference on Artificial Intelligence Circuits and Systems (AICAS), Incheon, Republic of Korea, 13–15 June 2022; pp. 371–374.
36. Tang, G.; Vadivel, K.; Xu, Y.; Bilgic, R.; Shidqi, K.; Detterer, P.; Traferro, S.; Konijnenburg, M.; Sifalakis, M.; van Schaik, G.J.; et al. SENECA: Building a fully digital neuromorphic processor, design trade-offs and challenges. *Front. Neurosci.* **2023**, *17*, 1187252. [[CrossRef](#)] [[PubMed](#)]
37. Orchard, G.; Frady, E.P.; Rubin, D.B.D.; Sanborn, S.; Shrestha, S.B.; Sommer, F.T.; Davies, M. Efficient neuromorphic signal processing with loihi 2. In Proceedings of the 2021 IEEE Workshop on Signal Processing Systems (SiPS), Coimbra, Portugal, 19–21 October 2021; pp. 254–259.
38. Davies, M. Taking neuromorphic computing to the next level with Loihi2. *Intel Labs’ Loihi* **2021**, *2*, 1–7.
39. Lin, C.K.; Wild, A.; Chinya, G.N.; Cao, Y.; Davies, M.; Lavery, D.M.; Wang, H. Programming Spiking Neural Networks on Intel’s Loihi. *Computer* **2018**, *51*, 52–61. [[CrossRef](#)]
40. Mosier, D.R. CHAPTER 1—Clinical Neuroscience. In *Neurology Secrets*, 5th ed.; Rolak, L.A., Ed.; Mosby: Philadelphia, PA, USA, 2010; pp. 7–17, ISBN 978-0-323-05712-7. [[CrossRef](#)]
41. Fusi, S.; Mattia, M. Collective behavior of networks with linear (VLSI) integrate-and-fire neurons. *Neural Comput.* **1999**, *11*, 633–652. [[CrossRef](#)]

42. Caporale, N.; Dan, Y. Spike timing-dependent plasticity: A Hebbian learning rule. *Annu. Rev. Neurosci.* **2008**, *31*, 25–46. [[CrossRef](#)]
43. Susman, L.; Brenner, N.; Barak, O. Stable memory with unstable synapses. *Nat. Commun.* **2019**, *10*, 4441. [[CrossRef](#)] [[PubMed](#)]
44. Pei, J.; Deng, L.; Song, S.; Zhao, M.; Zhang, Y.; Wu, S.; Wang, G.; Zou, Z.; Wu, Z.; He, W.; et al. Towards artificial general intelligence with hybrid Tianjic chip architecture. *Nature* **2019**, *572*, 106–111. [[CrossRef](#)] [[PubMed](#)]
45. Knight, J.C.; Tully, P.J.; Kaplan, B.A.; Lansner, A.; Furber, S.B. Large-scale simulations of plastic neural networks on neuromorphic hardware. *Front. Neuroanat.* **2016**, *10*, 37. [[CrossRef](#)] [[PubMed](#)]
46. Painkras, E.; Plana, L.A.; Garside, J.; Temple, S.; Galluppi, F.; Patterson, C.; Lester, D.R.; Brown, A.D.; Furber, S.B. SpiNNaker: A 1-W 18-core system-on-chip for massively-parallel neural network simulation. *IEEE J. Solid-State Circuits* **2013**, *48*, 1943–1953. [[CrossRef](#)]
47. Höppner, S.; Yan, Y.; Dixius, A.; Scholze, S.; Partzsch, J.; Stolba, M.; Kelber, F.; Vogginger, B.; Neumärker, F.; Ellguth, G.; et al. The SpiNNaker 2 processing element architecture for hybrid digital neuromorphic computing. *arXiv* **2021**, arXiv:2103.08392.
48. Deng, L.; Wang, G.; Li, G.; Li, S.; Liang, L.; Zhu, M.; Wu, Y.; Yang, Z.; Zou, Z.; Pei, J.; et al. Tianjic: A unified and scalable chip bridging spike-based and continuous neural computation. *IEEE J. Solid-State Circuits* **2020**, *55*, 2228–2246. [[CrossRef](#)]
49. Yu, L.; Chu, T.; Zhao, Z.; Mi, Y.; Yang, Y.; Wu, S. Spiking continuous attractor neural networks with spike frequency adaptation for anticipative tracking. In Proceedings of the 2019 IEEE International Workshop on Future Computing (IWOF), Hangzhou, China, 14–15 December 2019; pp. 1–3.
50. Neckar, A.; Fok, S.; Benjamin, B.V.; Stewart, T.C.; Oza, N.N.; Voelker, A.R.; Eliasmith, C.; Manohar, R.; Boahen, K. Braindrop: A mixed-signal neuromorphic architecture with a dynamical systems-based programming model. *Proc. IEEE* **2018**, *107*, 144–164. [[CrossRef](#)]
51. Moradi, S.; Qiao, N.; Stefanini, F.; Indiveri, G. A scalable multicore architecture with heterogeneous memory structures for dynamic neuromorphic asynchronous processors (DYNAPs). *IEEE Trans. Biomed. Circuits Syst.* **2017**, *12*, 106–122. [[CrossRef](#)] [[PubMed](#)]
52. Indiveri, G.; Corradi, F.; Qiao, N. Neuromorphic architectures for spiking deep neural networks. In Proceedings of the 2015 IEEE International Electron Devices Meeting (IEDM), Washington, DC, USA, 7–9 December 2015; pp. 2–4.
53. Cramer, B.; Krefl, M.; Billaudelle, S.; Karasenko, V.; Leibfried, A.; Müller, E.; Spilger, P.; Weis, J.; Schemmel, J.; Muñoz, M.A.; et al. Autocorrelations from emergent bistability in homeostatic spiking neural networks on neuromorphic hardware. *Phys. Rev. Res.* **2023**, *5*, 033035. [[CrossRef](#)]
54. Pershin, Y.V.; Slipko, V.A. Dynamical attractors of memristors and their networks. *Europhys. Lett.* **2019**, *125*, 20002. [[CrossRef](#)]
55. Wang, Y.; Yu, L.; Wu, S.; Huang, R.; Yang, Y. Memristor-Based Biologically Plausible Memory Based on Discrete and Continuous Attractor Networks for Neuromorphic Systems. *Adv. Intell. Syst.* **2020**, *2*, 2000001. [[CrossRef](#)]
56. Wei, Q.; Gao, B.; Tang, J.; Qian, H.; Wu, H. Emerging Memory-Based Chip Development for Neuromorphic Computing: Status, Challenges, and Perspectives. *IEEE Electron Devices Mag.* **2023**, *1*, 33–49. [[CrossRef](#)]
57. Ielmini, D.; Milo, V. Brain-inspired memristive neural networks for unsupervised learning. In *Handbook of Memristor Networks*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 495–525.
58. Milo, V.; Pedretti, G.; Laudato, M.; Bricalli, A.; Ambrosi, E.; Bianchi, S.; Chicca, E.; Ielmini, D. Resistive Switching Synapses for Unsupervised Learning in Feed-Forward and Recurrent Neural Networks. In Proceedings of the 2018 IEEE International Symposium on Circuits and Systems (ISCAS), Florence, Italy, 27–30 May 2018; pp. 1–5.
59. Davies, M.; Wild, A.; Orchard, G.; Sandamirskaya, Y.; Guerra, G.A.F.; Joshi, P.; Plank, P.; Risbud, S.R. Advancing Neuromorphic Computing with Loihi: A Survey of Results and Outlook. *Proc. IEEE* **2021**, *109*, 911–934. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.