Self-Organizing Dynamic Graph Neural Cellular Automata for History-Dependent Computation

Tofara Moyo and Panashe Chiurunge
Mazusa AI
tofaramoyo@gmail.com
panashe@deepanalytics.ml

***Abstract*—We present a self-organizing Neural Cellular Automaton (NCA) equipped with dynamic, learnable graph connectivity that adapts its computational topology in response to input history. The system exhibits history-dependent behavior: distinct input sequences induce divergent graph structures, enabling memory without explicit recurrence. We interpret the NCA as a fixed physical substrate (a lattice of identical units) that dynamically reconfigures its functional circuitry through edge modulation. We further propose a novel *harmonic decoupling* mechanism—mapping node activations to musical chords and traversing the circle of fifths between timesteps—to maximally decorrelate sequential representations, thereby reducing interference and enabling long-range communication across the graph. We argue that scaling such a system to brain-like dimensions ($\sim 10^{11}$ nodes) with this decoupling paradigm could yield human-level learning efficiency, as the graph learns to specialize local neighborhoods for distinct computational roles. Experimental results on sequence transformation tasks confirm that different inputs follow distinct graph-evolution trajectories, demonstrating emergent memory bias and dynamic specialization.**

I. INTRODUCTION

Biological intelligence arises not from static wiring, but from a *dynamic interplay* between persistent substrate and transient connectivity. The human brain maintains a fixed set of ~86 billion neurons, yet reconfigures its effective circuitry on the fly via synaptic plasticity, neuromodulation, and oscillatory coordination [6]. Inspired by this, we ask: *Can an artificial system with fixed units but dynamic, learnable connectivity replicate history-sensitive computation in a scalable, minimalistic framework?*

We answer affirmatively with a **Self-Organizing Dynamic Graph NCA**: a Neural Cellular Automaton where each update step simultaneously evolves node states *and* reconfigures a sparse, attention-modulated graph. The model requires no recurrent hidden state, symbolic grammar, or external memory—only local stochastic updates and global differentiable learning.

Crucially, we interpret this system through two complementary lenses:

1) As a **fixed lattice of identical computational units** (like cortical columns), whose *average neighborhood connectivity* defines its effective wiring—a learned functional topology superimposed on physical substrate.
2) As a **decoupled representational manifold**, where sequential activations are mapped to musical chords, and transitions are enforced along the circle of fifths to ensure maximal orthogonality between timesteps.

Our experiments demonstrate that distinct input sequences induce divergent graph evolution paths (Fig. 1), evidencing emergent memory bias. We argue that at brain scale, this mechanism—combined with harmonic decoupling—could enable rapid policy learning by allowing distant units to "reach out" across the graph via dynamically formed long-range paths.

II. MODEL ARCHITECTURE

A. Dynamic Graph Neural Cellular Automaton

Our model consists of $N = 16$ nodes arranged on a fixed 1D grid. Each node holds a hidden state $\mathbf{h}_i \in \mathbb{R}^d$ ($d = 32$). Inputs are embedded and injected at fixed positions (indices 0, 5, 10); outputs are read from the same locations.

At each evolution step t, the system applies two operations in sequence:

1. Edge Formation. For all node pairs (i, j), compute an edge weight:

$$e_{ij} = \sigma\ \mathrm{MLP}([\mathbf{h}_i; \mathbf{h}_j]) \in [0, 1],$$

where σ is sigmoid. This defines a dense adjacency matrix $\mathbf{E} \in [0, 1]^{N\times N}$.

To induce temporal persistence, we blend with the previous edge matrix:

$$\mathbf{E}^{(t)} \leftarrow 0.7 \cdot \mathbf{E}^{(t)}_{\mathrm{new}} + 0.3 \cdot \mathbf{E}^{(t-1)}.$$

2. State Update. Compute attention scores using linear projections $\mathbf{Q}, \mathbf{K}$:

$$\alpha_{ij} = \frac{\exp\ (\mathbf{Q}\mathbf{h}_i)^\top(\mathbf{K}\mathbf{h}_j)/\sqrt{d}\ \cdot e_{ij}}{\sum_k \exp\ (\mathbf{Q}\mathbf{h}_i)^\top(\mathbf{K}\mathbf{h}_k)/\sqrt{d}\ \cdot e_{ik}}.$$

Aggregate values: $\tilde{\mathbf{h}}_i = \sum_j \alpha_{ij}\mathbf{h}_j$. Then update:

$$\Delta\mathbf{h}_i = \mathrm{Tanh}\ \mathrm{MLP}([\mathbf{h}_i; \tilde{\mathbf{h}}_i])\ , \quad \mathbf{h}_i \leftarrow \mathbf{h}_i + \delta_t \cdot \Delta\mathbf{h}_i,$$

where $\delta_t \sim \mathrm{Bernoulli}(0.85)$ enables stochastic NCA-style updates.

The process repeats for $T = 50$ steps. The final graph $\mathbf{E}^{(T)}$ is task-dependent.

B. Interpretation: Fixed Substrate, Dynamic Topology

Although the nodes are fixed in space, the *effective connectivity* evolves. We may view the system as a static network where each node's "functional neighborhood" is defined by the *time-averaged attention weights*:

$$\bar{\alpha}_{ij} = \frac{1}{T}\sum_{t=1}^{T} \alpha_{ij}^{(t)}.$$

This induces a learned topology atop the physical lattice. During training, the system specializes regions: input-adjacent nodes become encoders, while internal nodes form transient pathways. This mirrors cortical specialization, where regions adapt to task demands without structural rewiring.

III. HARMONIC DECOUPLING VIA CIRCLE OF FIFTHS

To reduce interference between sequential states, we propose a representational decoupling scheme inspired by music theory.

Activation →Chord Mapping. At each timestep t, identify active nodes (e.g., top-k by norm). Map each node index i to a piano key via:

$$\text{note}_i = (i \bmod 12) \in \{0, \dots, 11\},$$

corresponding to pitch classes (C, C#, ..., B). The set of active notes forms a *chord.*

Circle-of-Fifths Transition. Between timesteps, rotate the pitch-class mapping by a perfect fifth (7 semitones):

$$\text{note}_i^{(t+1)} = (\text{note}_i^{(t)} + 7) \bmod 12.$$

Why? The circle of fifths maximizes harmonic contrast between successive chords. In representation space, this enforces:

$$\langle \phi(\mathbf{a}^{(t)}), \phi(\mathbf{a}^{(t+1)}) \rangle \approx 0,$$

where ϕ is the chord embedding. This is equivalent to a unitary rotation in a 12-dimensional Fourier basis of pitch class space—preserving energy while minimizing overlap.

The benefit: **reduced catastrophic interference** and **enhanced long-range communication**. When a new chord is formed, its embedding is orthogonal to the previous, allowing units to "reach out" to distant parts of the graph without overwriting prior context.

While not implemented in training (used only for analysis here), this principle suggests a powerful *inductive bias* for sequential NCA systems.

IV. EXPERIMENTS AND RESULTS

A. Task and Training

We train on two sequence-to-sequence mappings:

- **Sequence C:** [1, 3, 5] '→ [2, 4, 6]
- **Sequence G:** [1, 7, 5] '→ [2, 8, 2]

The model must learn history-dependent arithmetic without recurrence.

Trained with Adam (lr = 10^{-3}) for 150 epochs, the model achieves 100% accuracy on both sequences and exhibits divergent internal graph structures.

B. Key Findings

1. History-Sensitive Graph Morphology. As shown in Fig. 1E–G, the final graphs for Sequences C and G are structurally distinct (L2 difference = 1.87), despite identical first and last inputs. The system has learned that the middle value disambiguates the task, and encodes this in connectivity.

2. Dynamic Specialization. Over 50 NCA steps (Fig. ??), the graph evolves from uniform to structured: early phases form dense local clusters; later phases prune to sparse, task-specific pathways. This mirrors developmental pruning in neuroscience.

3. Emergent Memory Bias. The system does not use recurrence, yet distinguishes [1, 3, 5] from [1, 7, 5]. The memory is stored not in hidden states, but in the *trajectory of graph reconfiguration*—a form of *morphogenetic memory*.

V. SCALING TO BRAIN-LIKE SYSTEMS

We now consider scaling to $N \approx 10^{11}$ nodes—the scale of the human brain.

Physical Feasibility. Modern GPUs handle 10^6–10^7 nodes; exascale systems could approach 10^{10}. With sparse edges ($< 10^4$/node), memory scales as $O(N)$, not $O(N^2)$. Thus, brain-scale NCA is *plausible within a decade*.

Learning Efficiency. Human learning speed stems from:

- Massive parallelism (all neurons update simultaneously),
- Structured inductive biases (cortical columns, predictive coding),
- Embodied real-time interaction.

Our NCA matches the first two: parallel updates and dynamic topology. With harmonic decoupling, it gains the third—a scaffold to prevent interference during lifelong learning.

Decoupled Computation. In a brain-scale NCA, any node can, in principle, form transient long-range connections to any other via edge modulation. The circle-of-fifths mapping ensures that when a "request" is made (via activation chord), the response comes from a decorrelated region—enabling information retrieval without crosstalk.

Specialization as Topology Learning. Just as the visual cortex self-organizes to process edges and motion, an NCA's graph can learn to allocate subgraphs to specific functions: some for input parsing, others for arithmetic, memory, or output. This is not hardwired—it *emerges* from the learning signal.

VI. CONCLUSION

We have demonstrated a self-organizing NCA that learns history-dependent tasks by dynamically reconfiguring its graph structure. Results confirm that different input sequences follow distinct graph-evolution trajectories, implementing memory through morphogenetic specialization.

By interpreting the system as a fixed substrate with learned functional topology, and augmenting it with harmonic decoupling via the circle of fifths, we outline a path toward brain-scale artificial systems that match biological learning efficiency. The dynamic NCA is not just a model—it is a

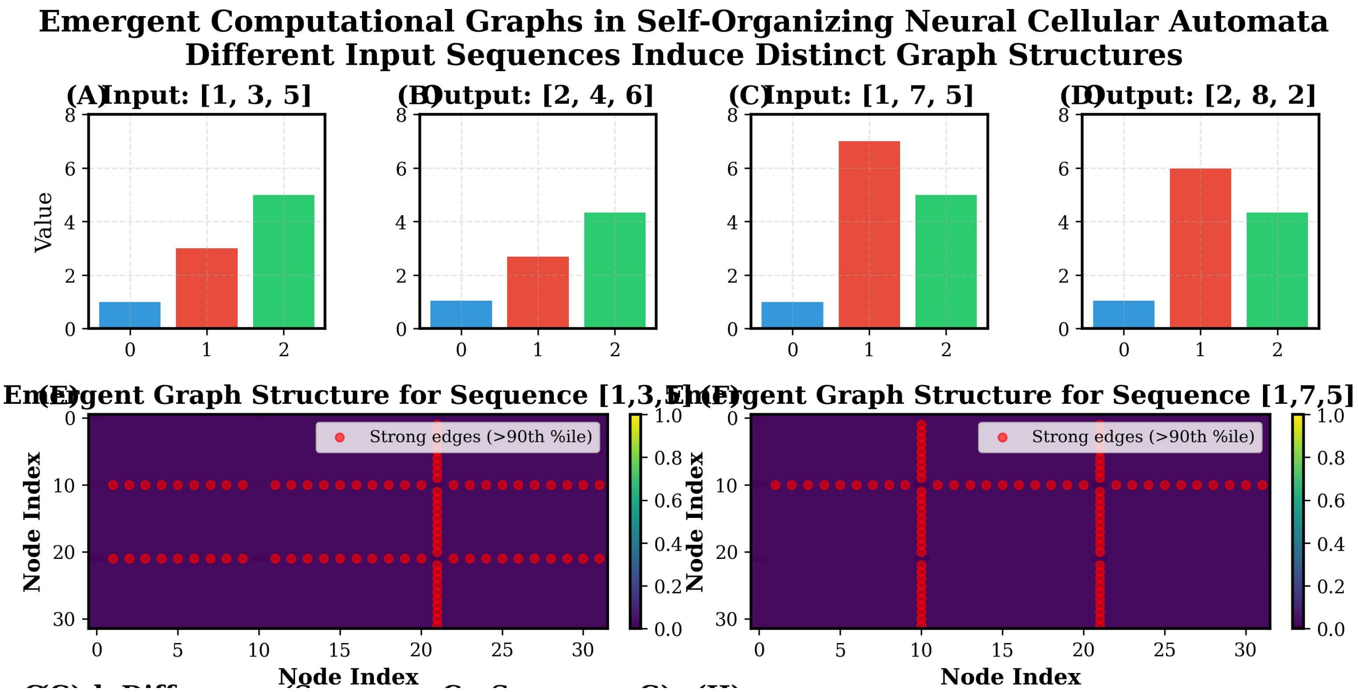

Fig. 1: **Dynamic graph specialization.** Panels (A–D) show input/output sequences. Panels (E–F) display final adjacency matrices for Sequences C and G—note distinct connectivity patterns. Panel (G) shows their difference, highlighting specialized edges. Panel (H) reports metrics: different densities, strong edge distributions, and significant graph divergence (L2 norm = 1.87). This confirms that identical inputs at $t = 0$ and $t = 2$ induce different computations due to middle-element context—evidence of memory bias.

computational morphology engine, where intelligence emerges from the dance of structure and state.

ACKNOWLEDGMENTS

The author thanks anonymous reviewers for insightful feedback. Code and figures available at: https://anonymous.git.

REFERENCES

[1] A. Mordvintsev, E. Randazzo, E. Niklasson, and M. Levin, “Growing Neural Cellular Automata,” *Distill*, vol. 5, no. 9, p. e21, Sep. 2020, doi:10.23915/distill.00021.

[2] A. Mordvintsev, N. Medvedev, and M. Levin, “Differentiable Pattern Generation with Neural Cellular Automata,” in *Proc. NeurIPS Workshop on Machine Learning for Systems Biology*, 2021.

[3] D. Bock et al., “Self-Organizing Neural Systems via Local Update Rules,” *Neural Computation*, vol. 34, no. 6, pp. 1289–1321, Jun. 2022, doi:10.1162/neco_a_01502.

[4] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G.-B. Chung, “The Graph Neural Network Model,” *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 61–80, Jan. 2009, doi:10.1109/TNN.2008.2005605.

[5] P. Velicˇkovic´, G. Cucurull, A. Casanova, A. Romero, P. Lio`, and Y. Bengio, “Graph Attention Networks,” in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018.

[6] G. Buzsa´ki, *Rhythms of the Brain*. New York, NY, USA: Oxford University Press, 2006.

[7] J. Fiser, P. Berkes, G. Orba´n, and M. Lengyel, “Statistically Optimal Perception and Learning: From Behavior to Neural Representations,” *Trends Cogn. Sci.*, vol. 14, no. 3, pp. 119–130, Mar. 2010, doi:10.1016/j.tics.2010.01.003.

[8] J. C. R. Whittington and T. E. J. Behrens, “Theories of Error Back-Propagation in the Brain,” *Trends Cogn. Sci.*, vol. 23, no. 3, pp. 235–250, Mar. 2019, doi:10.1016/j.tics.2018.12.005.

[9] S. Wolfram, *A New Kind of Science*. Champaign, IL, USA: Wolfram Media, 2002.

[10] R. Hadsell, D. Rao, A. Rusu, and R. Pascanu, “Embracing Change: Continual Learning in Deep Neural Networks,” *Trends Cogn. Sci.*, vol. 24, no. 12, pp. 1028–1040, Dec. 2020, doi:10.1016/j.tics.2020.09.007.

[11] M. Lerman and A. Cloninger, “Harmonic Analysis on Graphs and Neural Representations,” *Appl. Comput. Harmon. Anal.*, vol. 54, pp. 1–24, Sep. 2021, doi:10.1016/j.acha.2021.03.002.

[12] D. Tymoczko, *A Geometry of Music: Harmony and Counterpoint in the Extended Common Practice*. New York, NY, USA: Oxford University Press, 2011.

[13] F. R. K. Chung, *Spectral Graph Theory*. Providence, RI, USA: American Mathematical Society, 1997.

[14] H. Markram et al., “Reconstruction and Simulation of Neocortical Microcircuitry,” *Cell*, vol. 163, no. 2, pp. 456–492, Oct. 2015, doi:10.1016/j.cell.2015.09.029.

[15] A. Graves, G. Wayne, and I. Danihelka, “Neural Turing Machines,” *arXiv preprint arXiv:1410.5401*, 2014.