Petri Dish NCA

Ivy Zhang[1], Sebastian Risi[1], and Luke Darlow[1]

[1]Sakana AI, Japan
luke@sakana.ai

Abstract

Artificial life simulations in search of open-endedness are often based on artificial evolution. While such systems can generate complexity, gradient-based learning remains underexplored and may offer a complementary path to open-endedness. We introduce Petri Dish Neural Cellular Automata (PD-NCA): a differentiable multi-agent substrate consisting of a competitive population of neural cellular automata (NCA), trained continuously as an artificial life simulation. Differentiable PD-NCA enable end-to-end learning in a competitive, multi-agent system, which we hypothesize can induce open-ended complexification. Importantly, and unlike typical NCA experiments, the models in our experiments are continuously learning through gradient descent. Exploratory experiments demonstrate that PD-NCA show signs of emergent behavioral complexity and cooperation. More broadly, our work introduces a new substrate where learning could potentially combine with evolution to form open-ended systems.

Submission type: **Late Breaking Abstract**

Introduction

The field of artificial life (ALife) has seen much focus on evolutionary algorithms for optimization. Evolution holds significant potential for open-endedness [7], as it is not limited by the requirement for the underlying models, substrate, or techniques to be differentiable. That being said, recent work in differentiable systems have shown promise for ALife simulation [5, 8, 6, 1, 4].

Neural Cellular Automata (NCA) [5] are convolutional neural networks (CNNs) that learn local update rules, using backpropagation, to yield growth-like behavior in a 2D substrate. NCA demonstrated that backpropagation could learn complex local update rules, but were limited in two key ways: (1) they were single-agent-based, and (2) their loss functions were not designed for open-ended evolution.

We introduce Petri Dish NCA (PD-NCA), a framework that extends NCA to multi-agent scenarios where individuals compete for growth within a shared differentiable substrate. Unlike typical NCA experiments with fixed objectives, PD-NCA have no train-test split: models are continuously optimized throughout the simulation, making gradient-based learning part of the dynamics itself. This approach may prove more scalable than pure evolution, as backpropagation can efficiently optimize millions of parameters and benefits from neural scaling laws [2] that predict improved performance with increased model size.

In this paper we explore whether this combination of competition and continuous learning produces emergent ALife phenomena. We present our methods and initial evidence of spontaneous cooperation arising between competing agents. Our implementation is available **here** to enable further exploration by the ALife research community.

(a)
(b)
(c)

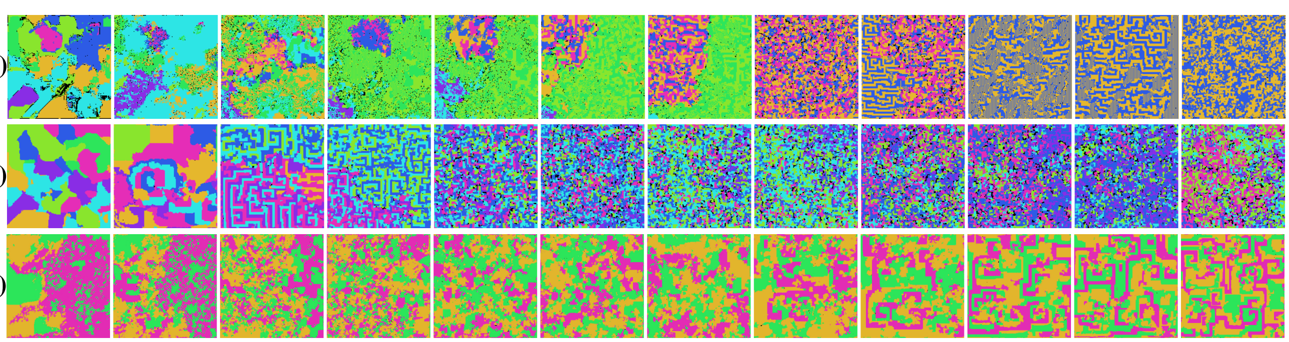

Figure 1: Signs of emergent complexity in PD-NCA: (a) structured pairing of two NCA showing signs of symbiosis, (b) stable continuous competition of several NCA, and (c) wave-like spiral patterns when using three NCA. These frames are of the simulation are not necessarily evenly spaced, but the arrow of time is from left to right. We share multiple videos **here**.

Methods

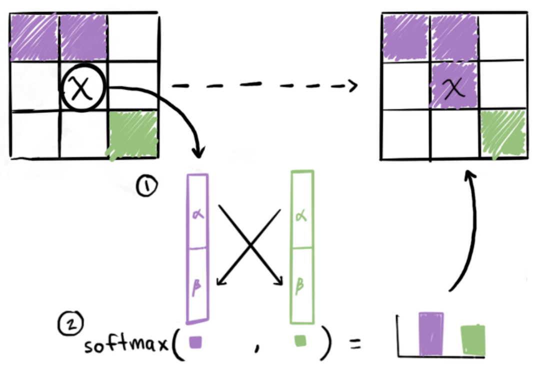

Figure 2: Competition mechanism between 2 NCA: (1) Cosine similarity between attack and defense are summed to form strength; (2) Strengths are passed through softmax to create relative update weight and new aliveness distribution.

Our simulation environment is a 2D grid where multiple NCA coexist and compete. A cell in the grid is defined by a state vector containing C channels, partitioned into 'attack', 'defense', and hidden channels. Fig. 2 shows how the attack and defense vectors interact between any two NCA. At each timestep, the simulation proceeds through three distinct phases: processing, competition, and state update.

Processing Each NCA is a CNN that 'views' a small region when proposing updates for a cell. These updates are masked by an aliveness value, allowing only updates to alive cells and their neighbors. Each NCA's aliveness is stored as a channel in the grid (hidden from the NCA). We also apply a static environment vector to provide a constant background update proposal, essentially bootstrapping growth and allowing for 'environmental' changes in future work.

Competition Proposed updates are resolved through a strength-based weighting system. At every cell, each NCA's strength is the sum of the cosine similarities between its attack channels and opposing defense channels. For example, consider a cell where NCA_A and NCA_B both propose updates. NCA_A's strength would be $\langle \text{att}_A, \text{def}_B \rangle + \langle \text{att}_A, \text{def}_{\text{env}} \rangle$, while NCA_B's strength would be $\langle \text{att}_B, \text{def}_A \rangle + \langle \text{att}_B, \text{def}_{\text{env}} \rangle$. Strengths are normalized via softmax to determine each NCA's relative contribution to the final update.

State update The final delta 'update' to the grid is the sum of the weighted proposed updates from each NCA and the background environment vector. The aliveness channels per cell are set to the relative strengths for each NCA. Fig. 3 shows the dominant aliveness at various snapshots throughout simulation. Any NCA with aliveness below a threshold has their aliveness redistributed among the remaining NCA.

NCA optimize for growth by maximizing their total aliveness across the grid. The result is that NCA grow outwards while also having to compete for space. We apply the logarithm function to an NCA's summed aliveness to stablize training. The result is that each NCA tries to minimize $L_i = -\log(\Sigma_{x,y} A_i(x, y))$; $A_i(x, y)$ is the NCA_i's aliveness at position (x, y).

Preliminary Results

We trained PD-NCAs using CNNs (up to 3 layers, 128 channels, $\approx 500K$ parameters), and up to 15 NCA on 256×256 grids. Fig. 1 shows selected frames from some simulations. Fig. 1(a) shows a simulation where structure emerges between groups of NCA (e.g., cyan-purple and blue-orange), giving credence to the potential of PD-NCAs as an ALife simulation. Fig. 1(b) and (c) both show patterns akin to chemical wave propagation and oscillatory dynamics. As further evidence, Fig. 3 shows the rise and fall of two distinct groups of NCAs.

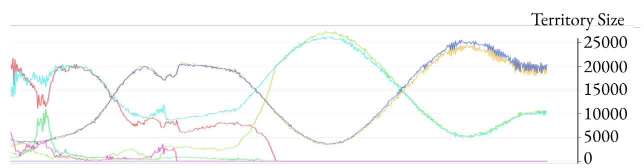

Figure 3: Territory size dynamics showing oscillations and cooperation between NCA pairs.

The complex dynamics of PD-NCA simulations are best understood through **video demonstrations** where structures emerge and disappear as the simulation unfolds. These dynamics arise because the underlying NCA are trained continuously throughout the simulation, learning to adapt *in-situ* as their environment changes. This adaptive behavior grew richer on larger grids, suggesting that scaling beyond 256×256 grids could yield even more complex dynamics.

Extensions

We plan to integrate evolution to create a hybrid system where learning and evolution operate simultaneously. For example, when NCA split spatially, the fragments could become independent lineages associated with their own optimizers. This would effectively lift the constraint on the number of alive NCA, thus enabling both gradient-based adaptation and evolutionary dynamics to work in tandem. Such a hybrid approach is designed to leverage the complementary strengths of gradient descent's efficiency when training CNNs and evolution's capacity for open-ended exploration.

Beyond individual growth objectives, we plan to explore system-wide optimization goals (e.g. compressibility) that leverage PD-NCA's fully differentiable architecture. Tools like ASAL [3] could automatically discover novel substrate parameterizations that produce even richer dynamics.

References

Etienne Guichard, Felix Reimers, Mia Kvalsund, Mikkel Lepperød, and Stefano Nichele. Arc-nca: Towards developmental solutions to the abstraction and reasoning corpus. *arXiv preprint arXiv:2505.08778*, 2025.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling Laws for Neural Language Models, January 2020. URL http://arxiv.org/abs/2001.08361. arXiv:2001.08361 [cs].

Akarsh Kumar, Chris Lu, Louis Kirsch, Yujin Tang, Kenneth O. Stanley, Phillip Isola, and David Ha. Automating the Search for Artificial Life with Foundation Models, May 2025. URL http://arxiv.org/abs/2412.17799. arXiv:2412.17799 [cs].

Pietro Miotti, Eyvind Niklasson, Ettore Randazzo, and Alexander Mordvintsev. Differentiable Logic Cellular Automata: From Game of Life to Pattern Generation, June 2025. URL http://arxiv.org/abs/2506.04912. arXiv:2506.04912 [cs].

Alexander Mordvintsev, Ettore Randazzo, Eyvind Niklasson, and Michael Levin. Growing neural cellular automata. *Distill*, 2020. doi: 10.23915/distill.00023.

Ettore Randazzo and Alexander Mordvintsev. Simulating an artificial biome of plants with biomaker ca. In *Artificial Life Conference Proceedings 36*, volume 2024, page 120. MIT Press One Rogers Street, Cambridge, MA 02142-1209, USA, 2024.

Kenneth O Stanley, Joel Lehman, and Lisa Soros. Open-endedness: The last grand challenge you've never heard of. *While open-endedness could be a force for discovering intelligence, it could also be a component of AI itself*, 2017.

Shyam Sudhakaran, Djordje Grbic, Siyan Li, Adam Katona, Elias Najarro, Claire Glanois, and Sebastian Risi. Growing 3d artefacts and functional machines with neural cellular automata. In *Artificial Life Conference Proceedings 33*, volume 2021, page 108. MIT Press One Rogers Street, Cambridge, MA 02142-1209, USA, 2021.