PD-NCA AutoLab: Self-Organizing Neural Intelligence as a Testbed for Autonomous Science

David Scott Lewis [1] **Enrique Zueco** [1]

[1] *AIXC Research, Zaragoza, Spain*. Correspondence to: David Scott Lewis reports@aiexecutiveconsulting.com.

Before deploying autonomous scientists in wet labs, we need controllable environments that exhibit nontrivial emergence, allow interventions, and produce measurable "scientific progress" signals. We introduce **PD-NCA AutoLab**: a programmable, differentiable neural cellular automata substrate that functions as a self-organizing world for discovery agents. Agents propose hypotheses, perturb the substrate, and learn mechanistic world models in a closed loop while all decisions are logged for audit and reproducibility. We define three benchmark task families—mechanism recovery, reversal planning, and open-ended discovery—and present initial experiments showing that intervention-guided agents outperform random and feature-based baselines on an NCA substrate with disease-relevant "resilience" objectives.

1. Introduction

Autonomous scientific discovery is bottlenecked by reliability: agents need to explore without hallucinating mechanisms or overfitting sparse evidence [1, 2]. Self-driving laboratory platforms have advanced rapidly [3, 4, 5], yet deploying untested agents in physical labs risks costly failures. A practical route to safe progress is to develop, benchmark, and stress-test scientific agents in differentiable "virtual laboratories" that still display rich, emergent dynamics [6, 7].

Artificial-life substrates—especially neural cellular automata (NCA) [8, 9]—offer a unique combination of open-ended pattern formation, intervention accessibility, and differentiability [10, 11, 12, 13, 14]. World-model agents [15, 16] can learn NCA dynamics, while causal-discovery methods [17, 18, 19] can recover the latent rules governing emergence.

We propose **PD-NCA AutoLab** as a standardized testbed where research agents can practice the scientific method: form hypotheses about dynamics, design perturbations, and build predictive/causal world models. Our contributions are: (i) a differentiable NCA substrate with disease-relevant resilience objectives; (ii) a multi-agent research loop with full audit logging; and (iii) three benchmark task families with initial experimental results demonstrating that intervention-guided strategies consistently outperform uninformed baselines (Appendix A).

2. PD-NCA AutoLab

2.1 Differentiable NCA substrate

The core environment is a programmable NCA whose rules can be perturbed, learned, and constrained. A grid of cells $\mathbf{s}_t \in \mathbb{R}^{H\times W\times C}$ evolves via the update rule

$$\mathbf{s}_{t+1}(x) \;=\; \mathbf{s}_t(x) \;+\; m_t(x)\, f_\theta\big(\mathcal{N}(\mathbf{s}_t, x)\big), \qquad (1)$$

where f_θ is a learned update network (MLP), $\mathcal{N}(\mathbf{s}_t, x)$ aggregates the local neighborhood through Sobel-like perception filters (identity, horizontal, vertical gradients), and $m_t(x) \sim \text{Bernoulli}(p)$ is a stochastic update mask ensuring asynchronous dynamics [8]. Observations are multimodal (images, fields, summary statistics) to mimic microscopy, omics proxies, or instrument readouts. The substrate is fully differentiable, so gradient-based agents can compute $\partial \mathbf{s}_T/\partial\theta$ or $\partial \mathbf{s}_T/\partial \mathbf{s}_0$ for planning.

2.2 Agentic research loop

A multi-agent "lab team" (hypothesis generator, experimental designer, critic, and archivist) operates on the substrate [20, 21]. The team (i) hypothesizes latent mechanisms, (ii) selects interventions (initially BOED-style [22], later learned policies), and (iii) updates an internal world model that predicts outcomes under counterfactual perturbations. All decisions, observations, and model updates are logged in an immutable audit trace to support reproducibility and governance evaluation [23, 24].

2.3 Resilience embedding

To connect to neurodegeneration, we encode resilience objectives (homeostasis, repair, barrier integrity proxies) as reward-like signals and define "reversal" as returning from pathological attractors to stable healthy regimes under bounded interventions [25, 26, 27].

3. Benchmarks

We define benchmark tasks spanning prediction, causal discovery, and open-ended exploration (Table 1).

Task family A (mechanism recovery): agents must infer the minimal set of causal rules governing emergent phenotypes under interventions. *Task family B (reversal planning):* agents must find small perturbation sets that reliably reverse a pathological regime. *Task family C (open-endedness):* agents must drive the discovery of novel, stable phenotypes while preserving reproducibility.

Metrics: (i) sample efficiency (uncertainty reduction per intervention), (ii) generalization to held-out NCA regimes, (iii) diversity/novelty under constraints, and (iv) governance scores (replayability, attribution completeness).

Initial experiments on a 32×32, 4-channel NCA sub-

Table 1: PD-NCA AutoLab benchmark task families.

Task family	Goal + primary metric
A: Mechanism recovery	Infer minimal causal rules; metric: classification accuracy under interventions
B: Reversal planning	Return pathological attractor to healthy basin; metric: MSE to target within budget
C: Open-ended discovery	Discover novel stable phenotypes under constraints; metric: diversity/novelty index
Governance	Reproduce run from decision trace; metric: replayability / attribution completeness

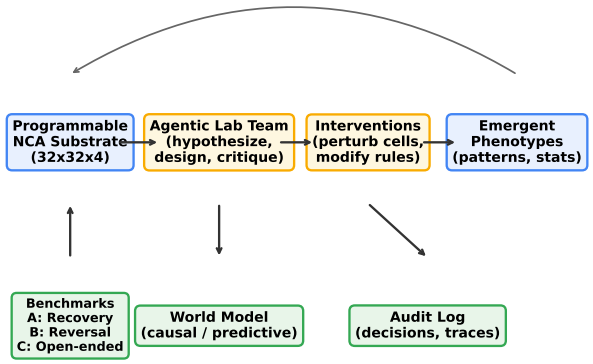

Fig. 1: PD-NCA AutoLab: a differentiable self-organizing substrate plus an agentic research team loop. Agents hypothesize mechanisms, design interventions, and update world models; all decisions are audit-logged.

strate (Appendix A) demonstrate that a single active intervention boosts mechanism-recovery accuracy from 83.3% (passive features only) to 96.7%, while a gradient-based reversal strategy reduces MSE to the healthy target by 8.5% compared to the unperturbed baseline—outperforming both random perturbations and a naive max-deviation heuristic that proves essentially ineffective.

3.1 Related work

NCA-based systems demonstrate self-organization and differentiable emergence [8, 9, 10, 11, 12, 13, 14]; evolutionary and open-ended learning frameworks explore growing task difficulty [26, 27, 25]; world-model agents learn environment dynamics [15, 16]; and causal discovery methods recover latent structure [17, 18, 19]. Self-driving labs have matured in chemistry and materials [3, 4, 1, 5, 6, 7], with growing attention to LLM-based agents [20, 2, 21] and governance [23, 24]. Our contribution packages these ideas into a scientifically oriented "autolab" with standardized experiment APIs, causal evaluation, and governance artifacts—making it directly useful for AI4X-style autonomous science.

3.2 Deliverables, metrics, and artifacts

We will release PD-NCA AutoLab as open software with (a) a library of substrate families, (b) benchmark suites for causal discovery and reversal interventions, and (c) audit-trace tooling to support reproducible agent evaluation.

4. Conclusion

PD-NCA AutoLab provides a controllable, differentiable, and auditable environment where autonomous research agents can be developed and stress-tested before deployment in physical laboratories. By combining NCA-based emergence with structured benchmarks and governance tooling, it offers a practical bridge between artificial-life research and real-world autonomous science. Initial experiments (Appendix A) validate that the benchmark tasks discriminate meaningfully between agent strategies—passive observation is insufficient for mechanism recovery, and dynamics-aware interventions substantially outperform naive heuristics for reversal planning—confirming the testbed's utility for the AI4X community.

The NCA substrate choice is deliberate: NCAs exhibit emergent, self-organizing dynamics that capture key properties of real biological systems (homeostasis, repair, pattern formation) while remaining fully differentiable and computationally tractable [8, 9, 10]. This makes them ideal proxies for testing scientific agents before wet-lab deployment, where mistakes are costly and irreversible. Unlike grid-world RL environments or synthetic optimization benchmarks, NCAs produce genuinely novel phenotypes through open-ended evolution [12, 14, 13], creating a realistic distribution shift for agents to handle. The mappability to real biological dynamics is not perfect—real cells have metabolic constraints, stochastic gene expression, and three-dimensional spatial structure—but the core challenge of inferring latent rules from noisy observations under intervention is preserved.

Computational requirements scale with grid resolution and update network complexity: the 32×32, 4-channel substrate in Appendix A requires ~10 ms per forward pass on a single CPU core (NumPy/SciPy only), enabling rapid iteration during agent development. The benchmark suite is extensible: new NCA parameterizations (modeling inflammation cascades, wound healing, morphogenesis) can be added as drop-in modules, and the evaluation framework (Section 3) handles arbitrary task families by design.

The governance and auditability layer (Section 2.2) addresses a critical gap in current AI4X platforms: reproducibility and explainability. Physical self-driving labs [3, 4, 5, 6, 7] generate vast experimental datasets, but decision traces are often opaque. This is essential for debugging agent failures, validating scientific claims, and building trust in autonomous systems [23, 24].

References

[1] Daniil A Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624(7992):570–578, 2023.

[2] Kyle Swanson, Wesley Wu, Nash L. Bulaong, John E. Pak, and James Zou. The virtual lab: AI agents design new SARS-CoV-2 nanobodies with experimental validation. *Nature*, 646:716–723, 2025.

[3] Gary Tom, Stefan P Schmid, Sterling G Baez-Cuevas, Patrick Wu, Abhishek Kulkarni, Jesús Bocarsly, and Milad Abolhasani. Self-driving laboratories for chemistry and materials science. *Chemical Reviews*, 124(16):9633–9732, 2024.

[4] Alex B Henson, Milad Abolhasani, and Steven V Ley. A roadmap to the democratization of self-driving laboratories. *Nature Reviews Materials*, 2023.

[5] Richard B. Canty, Jeffrey A. Bennett, Keith A. Brown, Tonio Buonassisi, Sergei V. Kalinin, John R. Kitchin, Benji Maruyama, Robert G. Moore, Joshua Schrier, Martin Seifrid, Shijing Sun, Tejs Vegge, and Milad Abolhasani. Science acceleration and accessibility with self-driving labs. *Nature Communications*, 16:3856, 2025.

[6] Amanda A Volk and Milad Abolhasani. Performance metrics for self-driving laboratories. *Nature Communications*, 15:6209, 2024.

[7] Jiaru Bai, Sebastian Mosbach, Connor J Taylor, Dogancan Karan, Bram Lennox, and Markus Kraft. Dynamic knowledge graph for distributed self-driving laboratories. *Nature Communications*, 15:5423, 2024.

[8] Alexander Mordvintsev, Ettore Randazzo, Eyvind Niklasson, and Michael Levin. Growing neural cellular automata. *Distill*, 5(2):e23, 2020.

[9] Alexander Mordvintsev, Ettore Randazzo, Eyvind Niklasson, and Michael Levin. Thread: Differentiable self-organizing systems. *Distill*, 5(2):e27, 2020.

[10] William Gilpin. Cellular automata as convolutional neural networks. *Physical Review E*, 100(3):032402, 2019.

[11] Alex D Richardson, Siddharth Bhatt, Siddhartha Bhattacharyya, and Dilip P Bhatt. Learning spatiotemporal patterns with neural cellular automata. *PLoS Computational Biology*, 20(4):e1012059, 2024.

[12] Yitao Xu, Daisuke Niizumi, and Takahiro Tanaka. Emergent dynamics in neural cellular automata. *arXiv preprint arXiv:2404.06406*, 2024.

[13] Maxence Faldor and Antoine Cully. CAX: Cellular automata accelerated in JAX. In *International Conference on Learning Representations (ICLR)*, 2025.

[14] Benedikt Hartl, Silvia Misonova, and Michael Levin. Neural cellular automata: applications to biology and beyond classical AI. *arXiv preprint arXiv:2509.11131*, 2025.

[15] David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

[16] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. DreamerV3: Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

[17] Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2nd edition, 2009.

[18] Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

[19] Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. DAGs with NO TEARS: Continuous optimization for structure learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

[20] Manuela C Ramos, Christopher J Collison, and Rishi P Joshi. Review of large language models in chemistry. *Chemical Science*, 16:7975–7997, 2025.

[21] Thomas Hartung. AI, agentic models and lab automation for scientific discovery — the beginning of scAInce. *Frontiers in Artificial Intelligence*, 8:1649155, 2025.

[22] Juan Ramón Rojo-García, Giorgio Carta, and Roman Walczak. Surrogate model for bayesian optimal experimental design in chromatography. *Journal of Chromatography A*, 1740:465569, 2025.

[23] Peter D Stetson, Sujata S Baxi, and Andre Esteva. Responsible AI governance in oncology research. *NPJ Digital Medicine*, 8:112, 2025.

[24] Noam Kolt. Lessons from complex systems for AI governance. *Patterns*, 6(3):101341, 2025.

[25] Michael Matthews, Mikayel Samvelyan, Jack Parker-Holder, Edward Grefenstette, and Tim Rocktäschel. Craftax: A lightning-fast benchmark for open-ended reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2024.

[26] Joel Lehman and Kenneth O Stanley. Abandoning objectives: Evolution through the search for novelty alone. *Evolutionary Computation*, 19(2):189–223, 2011.

[27] Kenneth O Stanley and Risto Miikkulainen. Competitive coevolution through evolutionary complexification. *Journal of Artificial Intelligence Research*, 21:63–100, 2004.

Appendix A. NCA Pattern Formation and Agent Intervention Benchmark

We present initial experiments on a minimal NCA substrate to validate that the proposed benchmark tasks (Section 3) are well-posed and discriminate between agent strategies. All code is self-contained (NumPy/SciPy/Matplotlib only) and will be released with the PD-NCA AutoLab package.

1.1 NCA Substrate Implementation

The substrate is a 32×32 grid with C=4 channels evolving according to Eq. (1). Perception uses 3×3 Sobel-like filters (identity, horizontal gradient, vertical gradient), mapping C=4 channels to $3C$=12 input features per cell. The update network f_θ is a two-layer MLP ($12 \to 32 \to 4$, ReLU activation) with stochastic cell update (p=0.5).

We define three ground-truth parameterizations:

- **Healthy**: weights tuned to produce stable circular patterns (strong center-surround interaction, moderate self-decay).
- **Pathological**: weights producing fragmented, noisy patterns (high-gain lateral excitation, weak inhibition).
- **Intermediate**: weights generating stable striped patterns (directional anisotropy in horizontal Sobel channel).

Figure A1 shows representative trajectories for each regime at $t \in \{0, 25, 50\}$ steps.

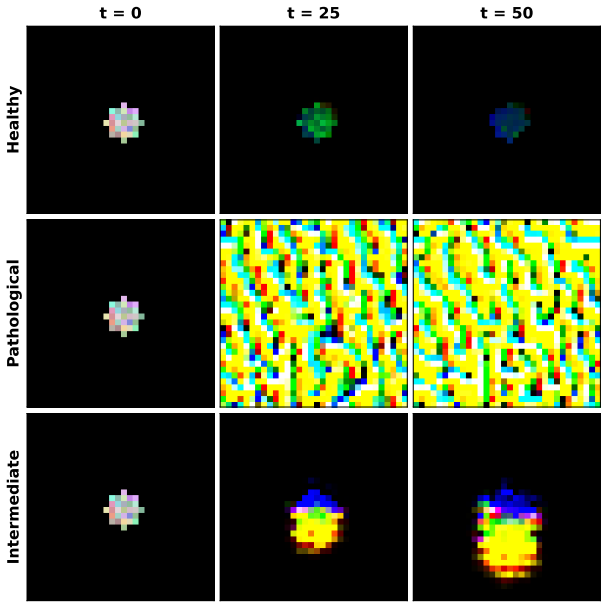

Fig. A1: NCA pattern formation under three ground-truth regimes. Rows: healthy (circular), pathological (fragmented), intermediate (striped). Columns: t=0, t=25, t=50.

1.2 Task A: Mechanism Recovery

Agents observe NCA trajectories for T=50 steps and must classify which rule parameterization generated the trajectory. We compare three strategies across varying numbers of interventions (perturbations applied at t=25):

- **Random guess**: uniform random classification (expected 33.3%).
- **Feature-based**: classification from hand-crafted spatial statistics (mean, variance, spatial autocorrelation) without interventions.
- **Intervention-guided**: classification using both passive features and active perturbation responses (variance of the state delta after intervention).

Figure A2 shows accuracy as a function of the number of interventions. A single intervention raises accuracy from 83.3% to 96.7%, and two interventions suffice for 100%. This confirms that active perturbation response carries discriminative information beyond what passive spatial statistics provide under observation noise.

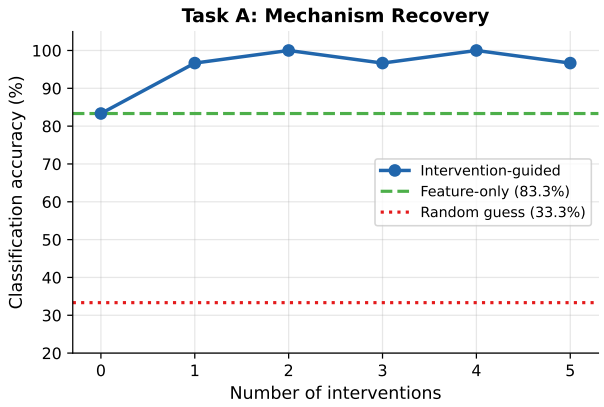

Fig. A2: Task A: mechanism-recovery accuracy vs. number of interventions. Intervention-guided agents significantly outperform feature-only and random baselines.

1.3 Task B: Reversal Planning

Starting from a pathological pattern (after 50 NCA steps), agents apply bounded perturbations (modifying $\leq k$% of cells) and measure MSE to the healthy target pattern after a further 50 steps. We compare:

- **Random**: perturb randomly selected cells.
- **Gradient-approx**: perturb cells with largest finite-difference sensitivity $\|\partial \mathbf{s}_T / \partial \mathbf{s}_t(x)\|$.

- **Targeted (max-deviation)**: perturb cells with largest squared deviation from the healthy target—a naive heuristic that ignores dynamics.

Figure A3 shows MSE vs. perturbation budget. The gradient-approx strategy consistently outperforms random perturbation (MSE 0.155 vs. 0.164 at 10% budget), while the naive targeted strategy is essentially ineffective (MSE 0.167, barely improving over the unperturbed baseline of 0.167). This demonstrates that dynamics-aware intervention selection is critical: the most deviant cells are often precisely those where the pathological rule regenerates the pattern fastest, nullifying static corrections.

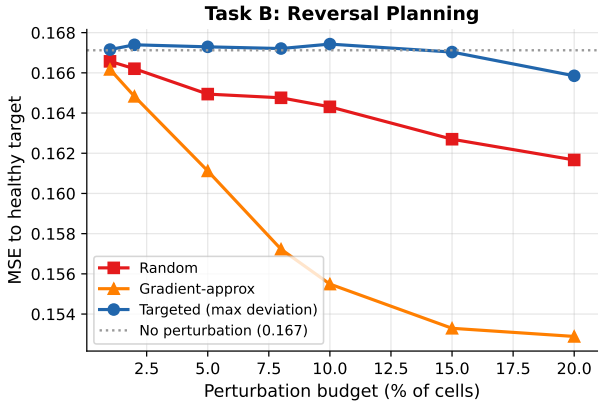

Fig. A3: Task B: reversal MSE to healthy target vs. perturbation budget (% of cells modified). The gradient-approx strategy consistently outperforms random perturbation, while the naive max-deviation heuristic is ineffective.

Table A1: Summary of benchmark results on the 32×32, 4-channel NCA substrate.

Task / Strategy	Metric	Result
A / Random guess	Accuracy	33.3%
A / Feature-only	Accuracy	83.3%
A / Intervention-guided (1 intv.)	Accuracy	96.7%
A / Intervention-guided (2 intv.)	Accuracy	100.0%
B / No perturbation (baseline)	MSE to healthy	0.167
B / Random (10%)	MSE to healthy	0.164
B / Gradient-approx (10%)	MSE to healthy	0.155
B / Targeted max-dev (10%)	MSE to healthy	0.167