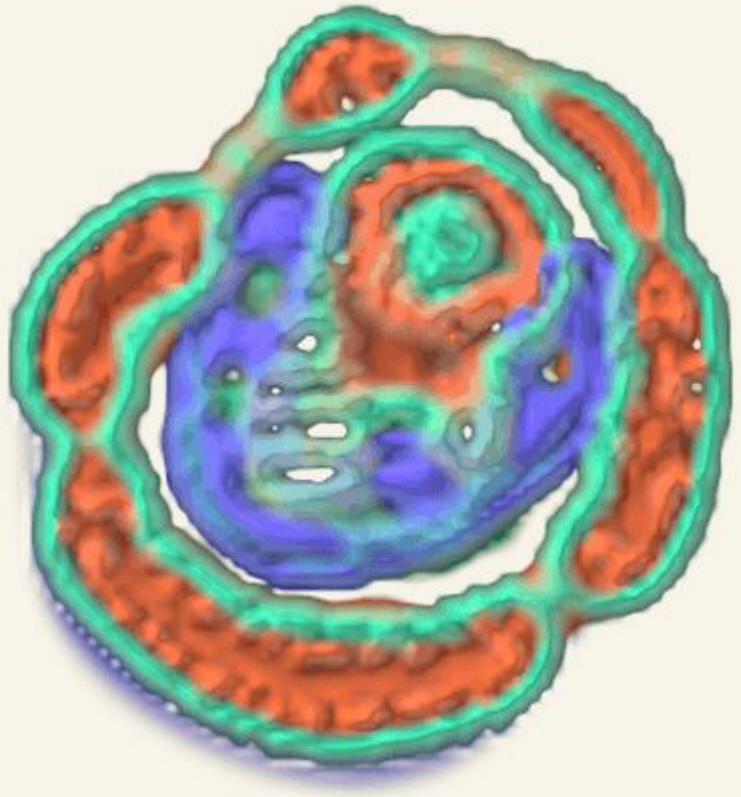
SPECTER LABS

Do different kinds of systems share principles of organization?



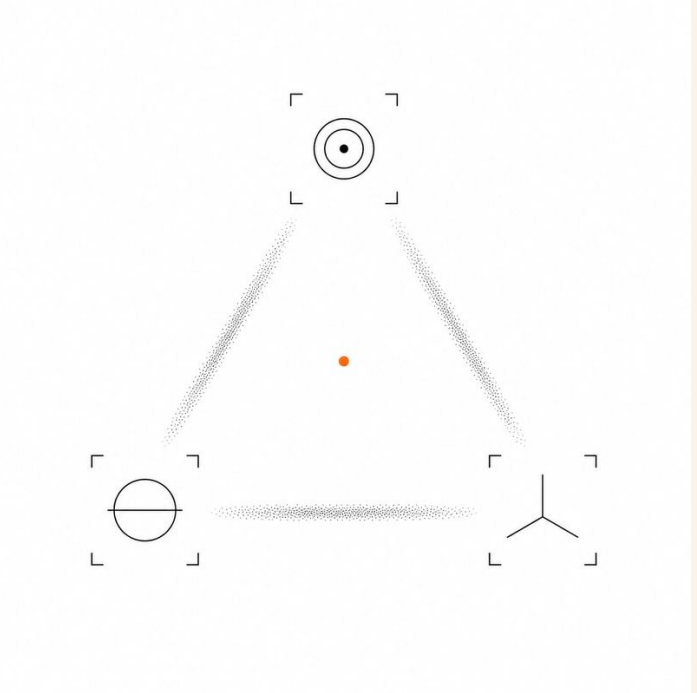
An introduction to our research for the Institute of Molecular Genetics and Genetic Engineering

Belgrade · October 2026

Specter Labs

We investigate whether different kinds of systems share principles of organization: how a body takes shape, how a collective finds a solution, how a pattern survives disruption.

We start from the picture Michael Levin lays out where embryos, machines and models act as interfaces to a space of patterns that is "structured" and "amenable to systematic exploration", a picture with older roots in D'Arcy Thompson and Thom on form, Kauffman on attractors and the adjacent possible, Hoffman, Fields and Prentner on interfaces, and Juarrero, Montévil and Mossio on constraints.



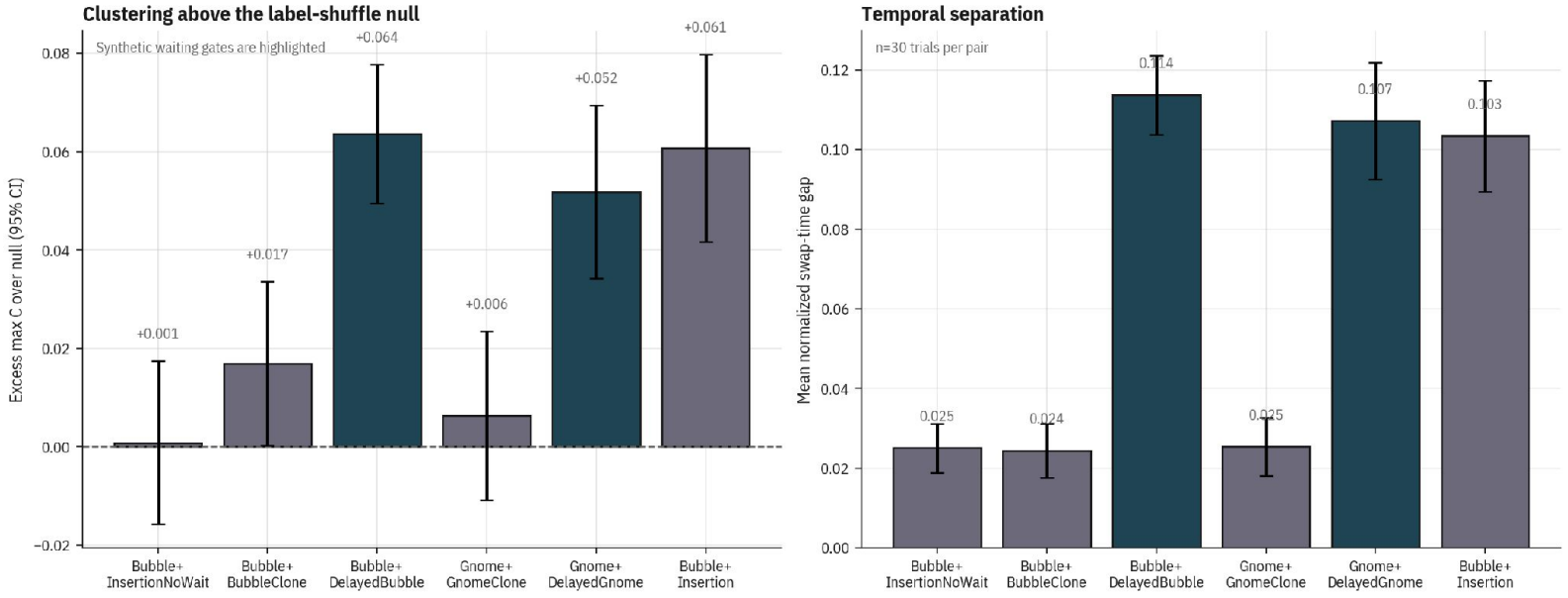
One assay across three systems so far

In each system we fix a goal, take something away, rerun under the same budget and classify what comes back.

System	What reaches a goal	The lesion	What we read
Sorting cells	cells that each run a local sorting rule	frozen cells, removed waiting gates	whether the array sorts, and how types of cells cluster
MCTS theorem proving	a prover closing a Lean theorem	one tactic family forbidden	replicate, reroute, collapse or rescue
Flow Lenia	a self-organizing body	moved, sheared or reassigned material	whether the body keeps its future, shape, behaviors

Competences in sorting algorithms

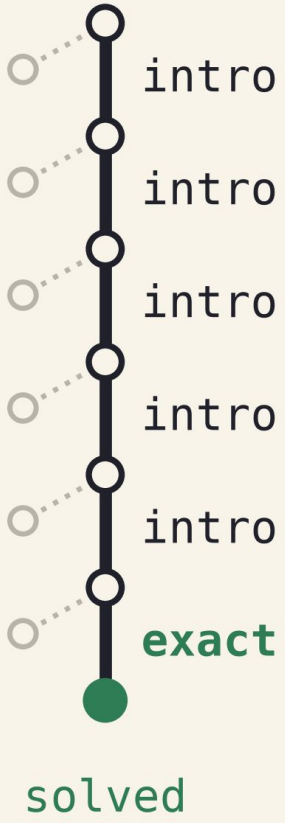
Zhang, Goldstein and Levin (2024) turned sorting algorithms into collectives of cells and reported competences nobody programmed: arrays route around damaged cells, temporarily giving up progress to do so ("delayed gratification"), and mixed cell types cluster. We traced each to a specific line of the algorithm. Getting around damage comes from one instruction to pick a new target when the old one is frozen, and clustering comes from Insertion cells waiting until everything to their left is sorted, so they move late and together.

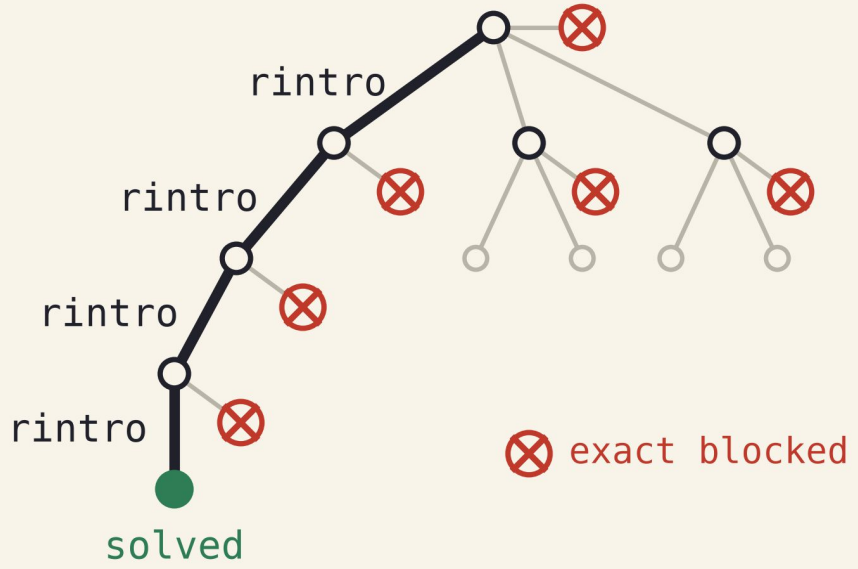


knocking out moves in a theorem prover

Following up in a way from Zhang and Levin damaging sorting cells to see what a collective still achieves, and we do the same to a theorem prover. A small language model suggests proof steps, a Monte Carlo tree search (the search behind AlphaGo) decides which to try, and Lean checks every step, so every proof that comes out is verified. We then take moves away from the prover (the way a geneticist knocks out genes, and watch how the search reorganizes). Some of what came back surprised us:

- Some knockouts make no difference at all, and others leave the prover almost unable to recover, like redundant and essential genes.
- The same theorem gets proved by many different routes, and no single route dominates.
- *Taking away cheap closing moves made the search faster.*
- Sometimes a knocked-out prover proves what the intact one could not, much like a suppressor mutation (preliminary).



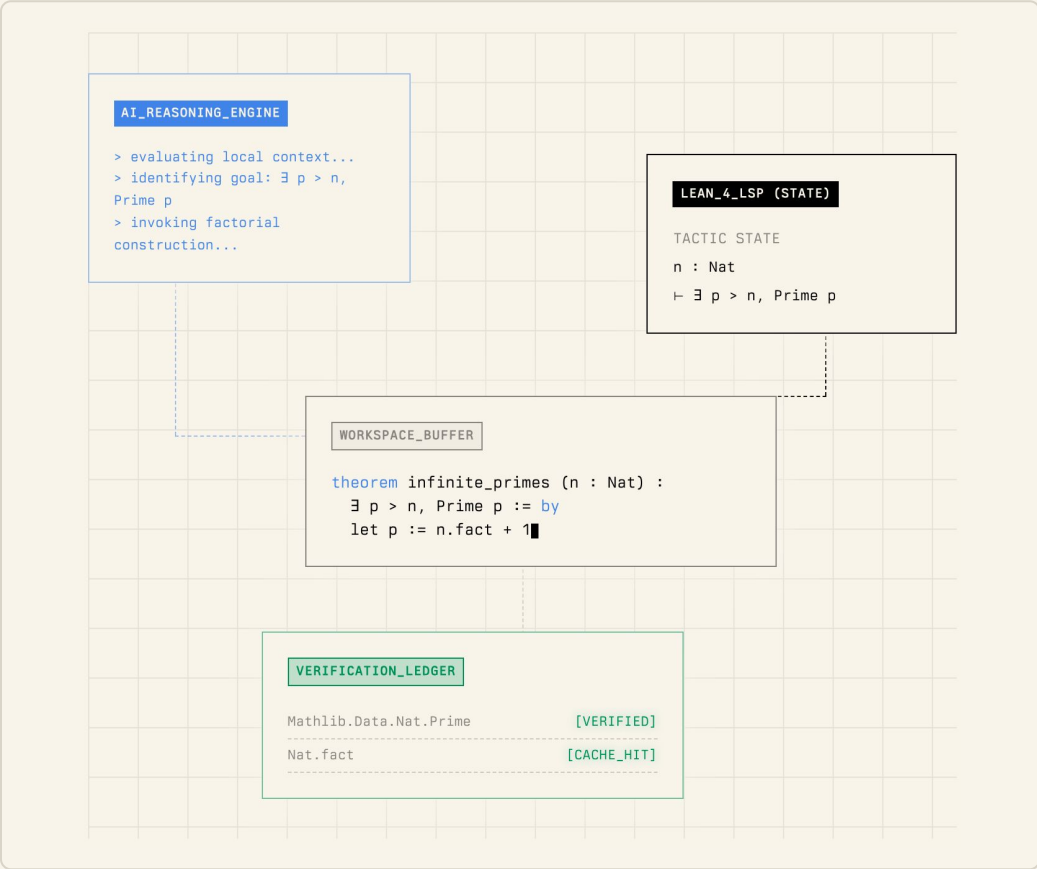


Apodeixis: infrastructure for checked mathematics

Apodeixis watches repositories of machine-checkable proofs in Lean, Coq and Isabelle, rechecks every commit in a clean environment, and signs a record of what ran and what the checker reported, so anyone can later confirm that a result was not altered.

People and AI agents claim open theorems, work on them and attach their prompts and logs as evidence, and each theorem carries a status of open, stated, blocked or proved. It works like continuous integration combined with a signed lab notebook.

apodeixis.specterlab.org



Formalizing Inter-universal Teichmüller Theory

Mochizuki's Inter-universal Teichmüller theory claims a proof of the abc conjecture. Scholze and Stix questioned it in 2018 one step has stayed in dispute (Theorem 3.11 to Corollary 3.12 of IUT III), with discussion largely stalled and acceptance mostly within Mochizuki's circle in Kyoto.

We are attempting to write that step out in Lean with every object and comparison included, to give mathematicians on both sides a shared place to work, where anyone can propose a missing argument or point to where one fails and have it checked, and, if the step holds, a route for the framework to be read and accepted beyond its current school. The goal is to support that debate rather than to produce volumes of machine-written proofs, and we take no side on whether IUT is correct.

~1.5M

lines of Lean across 773 modules

0

placeholders or extra axioms; every unproven input is an explicit hypothesis

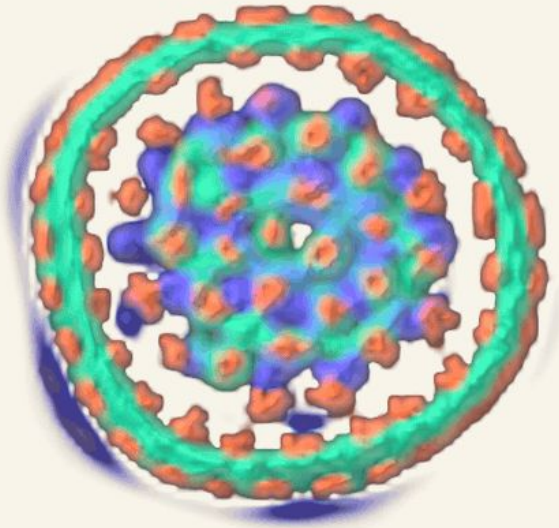
4

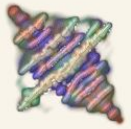
open propositions between Theorem 3.11 and Corollary 3.12

Flow Lenia: a model organism with its rulebook

Lenia is a continuous Game of Life whose few equations grow creatures nobody specifically designed: bilateral gliders, wheels, bodies that heal and divide, catalogued by species like radiolarians and jellyfish.

In Flow Lenia matter is conserved and each piece carries its own parameters, so species share one world and evolve (Plantec et al. 2023). Our bet is that memory and goal-seeking are generic properties of complex self-organizing dynamics, the attractors, multistability and hysteresis that cellular automata show in their simplest form.

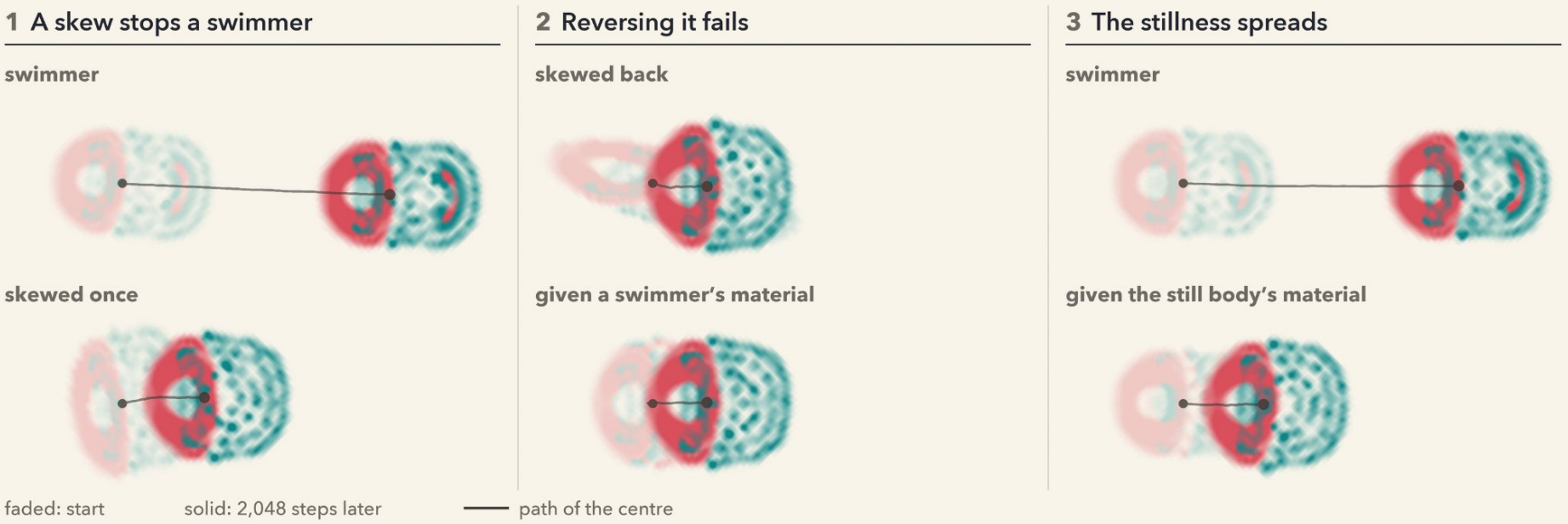






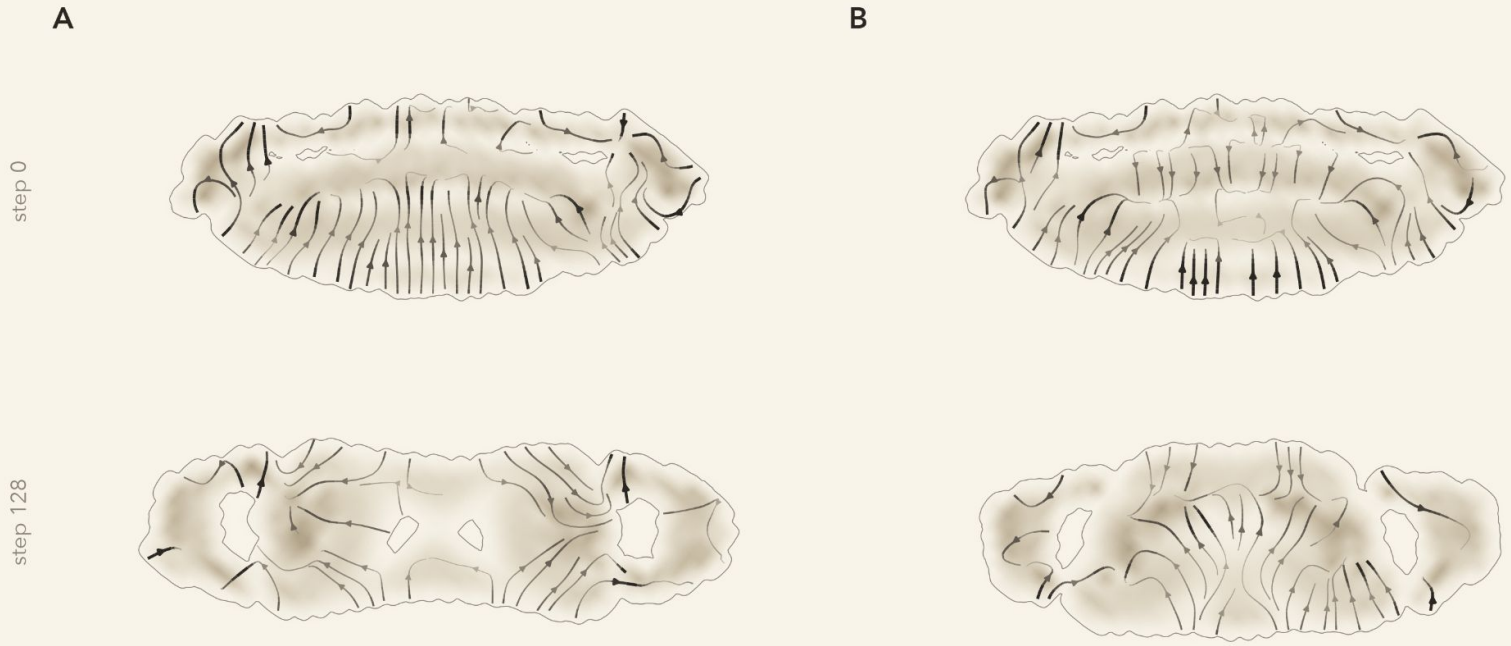
Memory in a Lenia Glider

In this experiment: we skew a gliding creature sideways once and it stalls for good. Nothing we tried restarts it, including skewing it back and giving it a glider's mix of materials, so the creature keeps a “memory” of the skew, as if its target state had been rewritten, much like planarians that keep regrowing two heads after a brief bioelectric shift (Durant et al. 2017). The stall is also *contagious* so to speak: give a glider a stalled glider's mix of materials and it stalls too.



Hidden Variations in Lenia

Two creatures can look identical pixel for pixel and still move differently from the first step, because their hidden / internal states differ. So a creature's shape says little about what it will do. Biology has the same thing in cryptic variation for example: flies carry hidden differences that only show up when the Hsp90 chaperone is weakened (Rutherford and Lindquist 1998). In Lenia we can measure how much hidden variation a body holds before its shape gives way.



Two bodies with the same outline but a different mix of materials carry different internal flows (lines), which pull them into different shapes within 128 steps.

Next: computation in living material

We are moving the same assay into organisms that solve problems without a nervous system and can be grown, perturbed and imaged cheaply, choosing each experiment so that a question from our simulations gets a living counterpart.

slime mold

Physarum polycephalum

A single cell that solves mazes with a network of tubes. Question: where does the network keep the memory of an old solution?

from Flow Lenia

social amoeba

Dictyostelium discoideum

Single cells that aggregate into a multicellular slug and sort into types. Question: does timing alone sort them?

from sorting cells

bacterial collectives

Biofilms

Colonies whose shape is built by growth under stress. Question: does the final structure record the order of the stresses?

from hidden state in Lenia

Further out

Can the dynamics of a body serve as its brain? A silicone octopus arm computes through the way it bends (Nakajima et al. 2015), and Physarum remembers food in the thickness of its tubes (Kramar and Alim 2021). A Lenia creature has no part that decides anything, so whatever it does, its body did. So, *which tasks can a body solve with nothing but its own dynamics?*

Can structures grown by biological collectives supply priors for machine learning? Wiring taken from real connectomes already improves networks (Lappalainen et al. 2024, Damicelli et al. 2022, Lechner et al. 2020), but these priors so far come only from neurons, and wiring grown by slime molds, amoebae, biofilms or cellular automata has not been tried.

Is there a structured space of patterns that bodies and behaviours come from? Raup (1966) showed that the coiled shells of snails, clams and ammonites fill only a small corner of the shapes possible on paper, and evolution favours forms that take little information to build (Johnston et al. 2022). Levin (2026) extends this beyond anatomy to behaviour and problem-solving, as a space of patterns that is "structured" and "amenable to systematic exploration". Lenia lets us explore it directly, by searching unrelated rules and asking whether the same bodies and the same behaviours keep appearing more often than simplicity alone would predict.

SPECTER LABS

Thank you

specterlab.org

contact@specterlab.org

