Evolutionary Implications of Multi-Scale Intelligence

Benedikt Hartl,[1,2] Sebastian Risi,[3] and Michael Levin[1,4,*)]
[1)] *Allen Discovery Center at Tufts University, Medford, MA, 02155, USA*
[2)] *Institute for Theoretical Physics and Center for Computational Materials Science (CMS), TU Wien, Austria*
[3)] *IT University of Copenhagen, 2300 Copenhagen, Denmark*
[4)] *Wyss Institute for Biologically Inspired Engineering at Harvard University, Boston, MA, 02115, USA*

(Dated: 9 April 2024)

ABSTRACT

In recent years, the scientific community has increasingly recognized the complex multi-scale competency architecture (MCA) of biology, comprising nested layers of active homeostatic agents, each forming the self-orchestrated substrate for the layer above, and, in turn, relying on the structural and functional plasticity of the layer(s) below. The question of how natural selection could give rise to this MCA has been the focus of intense research. Here, we instead investigate the effects of such decision-making competencies of an MCA's agential components on the process of evolution itself, using *in-silico* neuroevolution experiments of simulated, minimal developmental biology. We specifically model the process of morphogenesis with neural cellular automata (NCAs) and utilize an evolutionary algorithm to optimize the corresponding model parameters with the objective of collectively self-assembling a two-dimensional spatial target pattern (reliable morphogenesis). Furthermore, we systematically vary the accuracy with which an NCA's uni-cellular agents can regulate their cell states (simulating stochastic processes and noise during development). This allowed us to continuously scale the agents' competency levels from a direct encoding scheme (no competency) to an MCA (with perfect reliability in cell decision executions). We demonstrate that an evolutionary process proceeds much more rapidly when evolving the functional parameters of an MCA compared to evolving the target pattern directly. Moreover, the evolved MCAs generalize well toward system parameter changes and even modified objective functions of the evolutionary process. Thus, the adaptive problem-solving competencies of the agential parts in our NCA-based *in-silico* morphogenesis model strongly affect the evolutionary process, suggesting significant functional implications of the near-ubiquitous competency seen in living matter.

Keywords: Evolution, multi-scale competency, artificial intelligence, swarm intelligence, cells, embryos, development, self-assembly

[*)] **Corresponding author**: michael.levin@tufts.edu

GENERAL SUMMARY

Biological systems are composed of layers of organization, each level providing the foundation for the next higher level of abstraction: membranes, DNA, and proteins form cells, which then collectively organize into tissue, and, in further hierarchical steps, into tissues, organs, bodies, swarms, ecosystems, etc. Each of these layers has a degree of ability to adapt in real-time to new conditions to establish and maintain specific outcomes in terms of physiological, metabolic, transcriptional, and anatomical spaces. In other words, evolution works with material that is not passive matter but rather has a degree of competency – an agential material that forms the layer between the genotype and the phenotype. Many scientific studies have been dedicated to investigating how evolution gives rise to such intriguing problem-solving machines we call organisms. In this study, we ask the reverse question: what is it like to evolve over such a material, vs. one that passively maps genotypes into the form and function that selection operates over – how does it affect the process of evolution itself? We test this *in-silico* by utilizing evolutionary algorithms to adapt the behavior of a swarm of virtual uni-cellular agents in large-scale simulations of virtual embryos. In our minimal model, the cells collectively self-assemble a predefined target tissue on a neural cellular automaton. We find that competency at the cellular level of our multi-scale model system strongly affects the resulting evolutionary process, as well as the generalizability, evolvability, and transferability of the evolved solutions, suggesting profound evolutionary implications of the highly intricate multi-scale competency architecture of biological life.

I. INTRODUCTION

Biological systems are organized in an exquisite architecture of layers, including molecular networks, organelles, cells, tissues, organs, organisms, swarms, and ecosystems. It is well-recognized that life exhibits complexity at every scale. Increasingly realized however is the fact that those layers are not merely complex, but actually active "agential matter" which has agendas and competencies of its own[1,2]. Elsewhere, we have discussed examples of problem-solving in unconventional spaces, including transcriptional, physiological, metabolic, and anatomical space[3].

Especially interesting is the ability of these ubiquitous biological agents to deal with novel situations on the fly, which is not limited to brainy animals navigating 3D space, but also occurs with respect to injury, mutations, and other kinds of external and internal perturbations (reviewed in Ref. 4). One example of such problem-solving capabilities are the regenerative properties of some species that can regrow limbs, organs, or entire parts of their bodies when amputated, and - remarkably - stop when the precisely correct target morphology is complete[5–7]. This can be understood as cellular collectives navigating morphospace, until the desired target shape - or the goal - is reached again. Other examples include the ability of scrambled tadpole faces to reorganize in novel ways to result in normal frog faces[8], and the normal shape and size of structures in amphibia despite drastic changes in cell number[9] and cell size[10], which are handled by exploiting different molecular mechanisms to reach correct target morphologies despite novel changes in internal components. Behavioral and morphological plasticity intersect, in cases such as tadpoles made with eyes on their tails, which nevertheless can see and learn in visual assays without needing rounds of evolutionary adaptation[11].

The ability to navigate transcriptional and anatomical spaces, using perception-action loops and homeostatic setpoints, is now being increasingly targeted by biomedical and bioengineering efforts[12,13]. A fascinating body of work exists around the question of how neural and non-neural problem-solving capacities evolved, and how neuro-behavioral intelligence affects evolution[14–31]. However, we and others have previously suggested that somatic competency pre-dates neural intelligence[32–34], and has a bi-directional interaction with the evolutionary and developmental process[1,3,35]. Thus, here we address the second half of the evolution-intelligence spiral: how are evolutionary processes affected by the competency of the material? Especially important is the inclusion of the middle layer between genotype and phenotype. Mutation operates on genomes, and selection operates on phenotypic performance, but in most organisms, the connection between them is not linear or shallow - instead, developmental physiology provides a deep reservoir of dynamics that strongly alter the process. As a contribution to the study of evolvability and developmental mechanisms potentiating it[36–53], we established a virtual embryogeny[54] system focused on anatomical morphogenesis by cells. In this minimal model of morphogenesis, we were able to study the effects of different degrees of cellular competency on the evolutionary process.

The standard understanding of (Neo-Darwinian) evolution is schematized in fig. 1 (A): The *genome* of an organism encodes aspects of the organism's cellular hardware, which together define phenotypic traits. Given a competitive environment, natural selection then favors organisms with advantageous traits, and thus, on average, the corresponding genes tend to get passed on to the next generations more frequently. Random mutations may occur, consequently changing traits in the offspring phenotype. This affects the

offspring's reproductive success during the selection stage and, in that way, good traits prevail, and bad ones perish over time.

This view has been revised by, *e.g.*, Waddington[55,56], and more recent works[57–66], and has been the subject of vigorous debate[40,63,67–72] with respect to its capabilities for discovery, its optimal locus of control, and the degree to which various aspects are random (uncorrelated to the probability of future fitness improvements). Important open questions concern ways in which the properties of development – the layer between the mutated genotype and the selected phenotype – are evolved and in turn affect the evolutionary process[36,39,45,46,73–78]. Specifically, significant work has been done at the interface of evolution and learning - selectionist accounts of change and variational accounts of change respectively[30,61,62,66,79–85]. Significant progress has been made on the question of how evolution produces agents with behavioral competency in diverse problem spaces[17,86–88]. We have focused on a particular kind of competency – that of navigating anatomical morphospace[3,12,89,90]. More specifically, we here investigate *in silico* the evolutionary implications of the self-orchestrated process of morphogenesis, where local actions of single cells need to be aligned with a global policy of a multi-cellular collective to guide the formation of a large-scale tissue, in turn affecting the underlying evolutionary process. Work on developmental plasticity, chimeras, synthetic biobots, and the ability to overcome novel stressors has highlighted ways in which evolution seems to give rise to problem-solving machines, not fixed solutions to specific environments[91].

Thus, the problem-solving capacities of development, regeneration, and remodeling ensure that in many (perhaps most) kinds of organisms, the mapping from genotype to phenotype is not merely complex and indirect[92] (as schematized in fig. 1 (B)), but actually enables evolution to search the space of behavior-shaping signals, not microstates, and exploit modularity and triggers of complex downstream responses (c.f., fig. 1 (C,D)). We have previously argued that both evolution and human bioengineers face a range of unique problems and opportunities when dealing with the agential material of life – not passive or even just active matter, but a substrate that has problem-solving competencies and agendas at many scales[93,94]. What selection sees is not the actual quality of the genome, but the quality of the form and function of the flexible physiological "software" that runs on the genomically-specified molecular hardware, as schematically illustrated in fig. 1 (E). This in turn suggests that the actual progress of evolution should be significantly impacted by the degree and kind of competency in the developmental architecture. Prior work has suggested a powerful feedback loop between the evolution of morphogenetic problem-solving and the effects of these competencies on the ability of evolutionary search to produce adaptive complexity[1,35,95]. Here, we construct and analyze a new model of evolving morphogenesis, to study how different competency architectures within and among cells impact evolutionary metrics such as rate, robustness to noise, and transferability to new environmental challenges.

To quantitatively study the effects different levels of competency of the decision-making centers in a multi-scale competency architecture have on the process of evolution, we here rely on tools from the research field of Artificial Life[96] which furthers computational and cybernetic models that mimic life-like behavior based on ideas taken from biology; a simple example is cellular automata (CAs)[97]. In such CAs, the (numerical) states of localized cells, organized on a discrete, spatial grid, change in time via local update rules. Although typically rather simple "hardcoded" update rules are employed, CAs often display complex dynamics (c.f., Conway's *Game of Life*[98] or *Lenia*[99]) but are not known

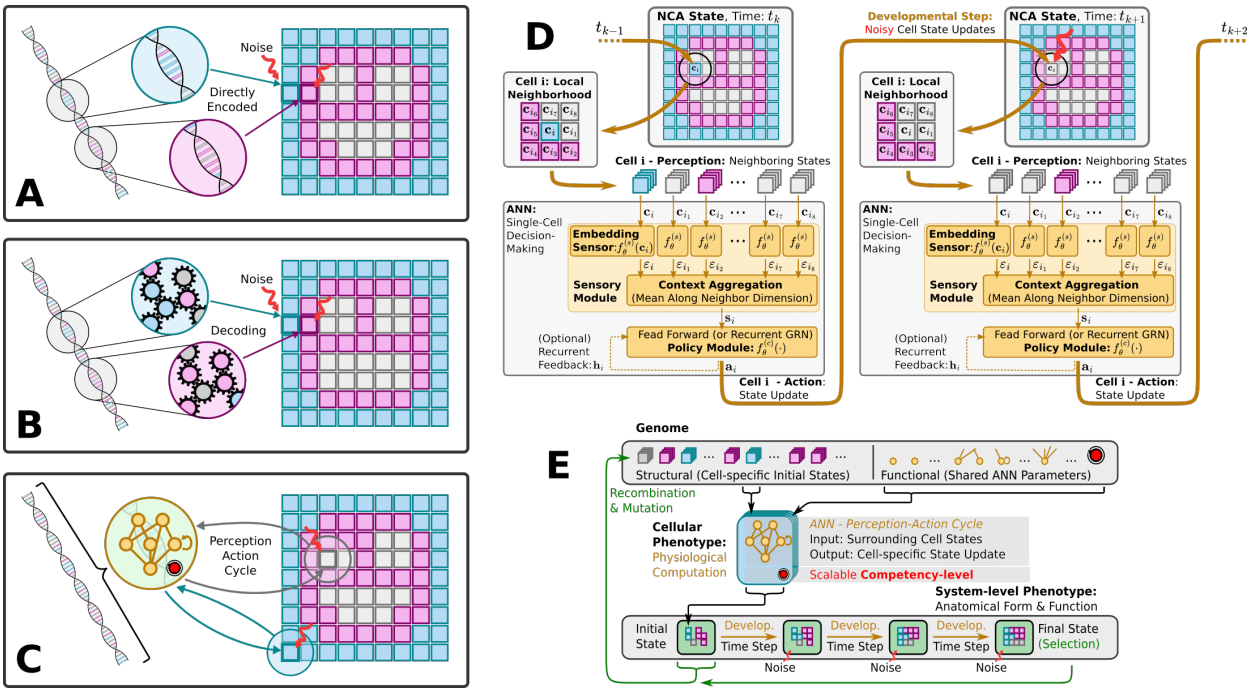

FIG. 1: (**A**-**C**): Illustration of different ways of genetic encodings of a phenotype of, here, a two-dimensional smiley-face tissue composed of single cells. (**A**) Direct encoding: Each gene encodes a specific phenotypic trait, here of each specific cell type of the tissue, colored blue, pink, and white. (**B**) Indirect encoding: A deterministic mapping between the genome and different phenotypic traits, here again of each cell type (shown for completeness, but not investigated here due to reasons discussed in section IV). (**C**) Multi-scale Competency Architecture: Encoding of functional parameters of the uni-cellular agents which self-assemble a target pattern via successive local perception-action cycles[1] (c.f., panel **D**). In all three panels, we schematically illustrate from left to right the genome, the respective encoding mechanism, and the corresponding phenotype; colors indicate cell types, and arrows indicate the flow of information and environmental noise, affecting each cell during the developmental process. (**D**) Detailed information-flow-chart of the perception-action cycle of a particular single cell agent, labeled i, in a Neural Cellular Automaton (NCA)-based multi-scale competency architecture (c.f., panel **C** and section II A): Starting from a multi-cellular phenotype configuration at time t_k (left smiley-face panel), and following the thick orange arrows, each cell i perceives cell state information about its respective local neighborhood of the surrounding tissue (respectively labeled). This input is passed through an artificial neural network (ANN), substituting the internal decision-making machinery of a single cell, until an action output is proposed that induces a (noisy) cell state update in the next developmental step at time t_{k+1} (details on labeled internal ANN operation and ANN architectures are introduced later in section II A and appendix A). (**E**) Schematic illustration - following Ref. 1 - of the evolution of a morphogenesis process with a multi-scale competency architecture acting as the developmental layer between genotypes and phenotypes (see sections II A and II B for details): The genotype (top) encodes the structural (initial cell states) and functional parts (decision-making machinery) of a uni-cellular phenotype (center). The cell's decision-making machinery is represented as a potentially recurrent ANN (yellow/orange graph) with an adjustable competency level (red knob). Through repeated local interactions (perception-action cycles; detailed in panel **D**)), the multi-cellular collective self-orchestrates the iterative process of morphogenesis and forms a final target pattern, *i.e.*, a system-level phenotype after a fixed number of developmental steps (bottom left to right) while being subjected to noisy cell state updates at each step (red arrows). The evolutionary process solely selects at the level of the system-level phenotypes (labeled *Final State* at the bottom right). Based on a phenotypic fitness criterion, the corresponding genotypes - composed of the initial cell states (bottom left) and the functional ANN parameters (top right) - are subject to evolutionary reproduction - recombination and mutation operations - to form the next generation of cellular phenotypes that successively "compute" the corresponding system-level phenotypes via morphogenesis, etc.

to exhibit homeostatic (closed-loop) activity. An extension of CAs, termed neural cellular automata (NCAs)[100], utilize artificial neural networks (ANNs) as more flexible trainable update rules, aiming to model the internal decision-making machinery of biological cells.

Employing machine learning methods, such NCAs have been trained to perform self-orchestrated pattern-formation (notably, of images from a single "seed" cell)[101] and even the co-evolution of a rigid robot's morphology and its controller has been demonstrated with such NCAs[102].

NCAs exhibit a striking resemblance with the genome-based multi-scale competency architecture of biological life[102], as illustrated in fig. 1 (C-E): an organism's entire building plan is encoded in its genome (corresponding to the NCA's parameters), while its cells collectively run the self-orchestrated developmental program of morphogenesis (realized by the NCA's layout and ANN architecture) via perception-action cycles at the uni-cellular level (cell state updates in the NCA, c.f., fig. 1 (D)). Starting from an initial cell state configuration of the NCA, the details of a virtual organism are then, step-by-step, "refined" in a collective self-organizing growth phase on the cellular level, and maintained against cell state errors later on in the virtual organism's lifetime. Thus, a single (trainable) NCA guides the growth and integrity of a virtual organism's tissue via intracellular information processing and intercellular communication, imitating *in silico* the multi-scale competency-based process of morphogenesis and morphostasis.

Here, we deploy a swarm of virtual uni-cellular agents on the spatial grid of an NCA. As illustrated in fig. 1 (D), each uni-cellular agent's internal decision-making machinery is modeled by an ANN that allows each agent to independently perceive the cell states of its adjacent neighbors on the grid and propose cell state update actions to regulate its own cell state over time. The collective of cells thereby forms a spatial pattern or tissue of cell states on the NCA via local communication rules.

We utilize evolutionary algorithms (EAs)[103] as simulated evolutionary process to optimize the parameters of such NCAs, so the uni-cellular agents evolve to collectively self-assemble a predefined target pattern of cell states in a fixed number of developmental steps; see fig. 1 (E) for a flow-chart of the evolutionary process. We explicitly separate the NCA parameters into a structural and a functional part. The structural parameters describe the initial cell state, and the functional parameters the weights and biases of the ANN of each agent, as illustrated by the "Genome" in fig. 1 (E). Both the structural and functional part of the genome are compiled into a swarm of uni-cellular phenotypes on the grid of the NCA. Thus, starting from an initial cell state configuration, given by the structural part of the genome, the NCA's uni-cellular agents run the developmental program of morphogenesis via successive perception-action cycles (see fig. 1 (D)) to self-assemble in successive developmental steps a system-level phenotype, *i.e.*, a two-dimensional pattern of cell states on the NCA. The deviation of these final cell state configurations from a desired target pattern - here, a Czech flag- or smiley-face-pattern reminiscent to that of the amphibian craniofacial prepattern[104] - defines the phenotypic fitness score of a particular NCA realization. Based on an entire population of NCAs, and on the corresponding fitness scores, the EA successively samples the genomes of the next generation of NCAs which, over time, evolve to reliably self-assemble the target pattern.

The conceptually simple process of cell state updates of NCAs and the ANN-based modelling of the uni-cellular decision-making allow us to interfere with (I) the reliability of the cell state update executions, and (II) with the computational capacity of the ANNs that guide each cell's decision-making. To vary the former (I), we introduce a decision-making probability, P_{D}, that specifies the probability at which a proposed update of each individual cell is executed in the environment (or omitted otherwise). Thus, by tuning the decision-making probability from $P_{\mathrm{D}} = 0$ to $P_{\mathrm{D}} = 1$, we can continuously vary the behavior of the

NCA from a direct-encoding scheme without competency to a multi-scale competency architecture with perfect reliability in cell state update executions.

To systematically vary the computational capacity of the involved ANNs (II), we introduce independent copies of a particular sub-module of the uni-cellular agents' ANNs, *i.e.*, of the policy module illustrated in fig. 1 (D) (see sections II A and III B and appendix A for details on the ANN architectures). This increases the number of trainable parameters of the ANNs which are responsible for performing the same operation, namely interpreting the cell's local environment and proposing a cell state update action. Thus, by taking the average output of all redundant policy-modules of a single agent, a cell's decision-making can be biased by the several redundant paths through which signals are transmitted in the ANN, inspired by error-correcting codes[105–107]. We explicitly define a redundancy number, R, that specifies how many redundant copies of the policy module are used in the ANNs of an NCA's cells.

The decision-making probability (I) and the redundancy number (II) represent two levers of competency in our system (schematically illustrated by the red know in fig. 1), which we can scale continuously (I) or discretely (II) to systematically tune the behavior of an NCA. Throughout the manuscript, we refer to these two parameters as "competency levels", but we would like to stress that many more options would have been possible to vary the competency in our system. For instance, the particular ANN architecture can have large effects on the competency of the uni-cellular agents; a systematic investigation thereof is out of the scope of this work. Here, we utilize two particular ANN architectures, one based on a feed forward (FF) and one based on a recurrent ANN architecture[108] that is inspired by gene regulatory networks (GRNs)[109], which we thus term recurrent gene regulatory network (RGRN), see appendix A for details.

To study the effects of different competency levels of the decision-making centers in a multi-scale competency architecture on the underlying evolutionary process of a morphogenesis task, we systematically vary in large-scale simulations the decision-making probability (I) and the redundancy number (II) of NCAs with FF and RGRN ANN architectures. Furthermore, we expose the corresponding NCAs to different noise conditions during cell state updates (III) and perform several statistically independent evolutionary searches at each parameter combination (I-III) to investigate the performance of the evolutionary process of finding solutions to such noisy pattern formation tasks.

The manuscript is organized as follows: In section II, we describe the numerical and computational methods applied herein. More specifically, we introduce NCAs in section II A, and describe the neuroevolution approach used to optimize the NCAs ANN parameters based on ideas of evolution and natural selection via EAs in section II B. We specify the particular morphogenetic problem we primarily focused on - the 8×8 Czech flag task - in section III A, and compare in section III B the efficiency of evolving the target pattern via a direct encoding scheme and a multi-scale competency architecture. In section III C, we functionally define and systematically vary the different tuneable competency levels in our system to illustrate the evolutionary implications of utilizing a multi-scale competency architecture rather than a direct encoding scheme for morphogenesis tasks. We then study the effects of allowing the evolutionary process to afford competency as a gene during optimization in section III D, and eventually investigate our multi-scale competency approach for robustness and generalizability regarding system parameter changes in section III E, and for transferability to modified target patterns in section III F. We conclude in section IV, and attach an Appendix.

II. METHODS

A. Neural Cellular Automaton: A Multi-Agent Model for Morphogenesis

Cellular Automata (CAs) have been introduced by von Neumann to study self-replicating machines[97] and are simple models for *Artificial Life*[96]. In CAs, a discrete spatial grid of cells is maintained over time, each cell i being attributed a binary, integer, real, or even vector-valued state, $\mathbf{c}_i(t_k)$, at each step in time t_k. The cell states evolve over time via local updated rules, $\mathbf{c}_i(t_{k+1}) = f_u(\mathcal{N}_i(t_k))$, as a function of its own, $\mathbf{c}_i(t_k)$, and the numerical states, $\mathbf{c}_{i_\nu}(t_k)$, of its $i_{\nu=1,\dots,N}$ neighboring cells on the grid, that we collect in the matrix $\mathcal{N}_i(t_k) = (\mathbf{c}_i(t_k), \mathbf{c}_{i_1}(t_k), \dots, \mathbf{c}_{i_N}(t_k))$. Although typically rather simple "hardcoded" (i.e., predefined) update rules $f_u(\cdot)$ are employed, CAs often display complex dynamics and can even be utilized for universal computation (c.f., Conway's *Game of Life*[98] or Wolfram's *rule 110*[110,111]).

Neural cellular automata (NCAs)[100] extend CAs by replacing the local update rule with more flexible[112] artificial neural networks (ANNs), $f_u(\cdot) \rightarrow f_\theta(\cdot)$, where θ denotes the set of trainable parameters of the ANN (see appendix A for details). Employing *Machine Learning*, such NCAs have been trained to perform self-orchestrated pattern-formation[101] (notably, of RGB-images from a single "seed" pixel) and even the co-evolution of a rigid robot's morphology and its controller has been demonstrated recently with NCAs *in silico*[102]. Such self-orchestrated pattern-formation is reminiscent of the self-regulated development of a biological organism, from a single fertilized egg cell to a complex anatomical form. Thus NCAs have been proposed as toy models for morphogenesis[101].

An NCA basically represents a grid of cells that are equipped with identical ANNs, each perceiving the numerical cell states of its host's local environment, $\mathcal{N}_i(t_k)$, and proposing actions, $\mathbf{a}_i(t_k) = f_\theta(\mathcal{N}_i(t_k))$, to regulate its own cell state

$$\mathbf{c}_i(t_{k+1}) = \mathbf{c}_i(t_k) + \mathbf{a}_i(t_k) + \xi_\mathbf{c}, \tag{1}$$

- and, in turn, the cell states of its neighbors - where we also account for potential noise $\xi_\mathbf{c}$ in the environment during the process of morphogenesis. Thus, each cellular agent can only perceive the numerical states of its direct neighbors, $\mathcal{N}_i(t_k)$, at an instant of time, t_k, and, in turn, communicate with these neighbors via cell state updates, $\mathbf{c}_i(t_{k+1})$, following a policy $\pi(\mathcal{N}_i(t_k)) \approx f_\theta(\mathcal{N}_i(t_k))$ that is approximated by an ANN with parameters θ. Through the lens of *Reinforcement Learning*[113], an NCA can thus be understood as a trainable, locally-communicating multi-agent system that can be utilized such that the collective of cells achieves a target system-level outcome (see appendix B for details).

In contrast to previous contributions of *in silico* morphogenesis experiments in NCAs[101], we here do not use standard convolutional filters in our ANN architectures but utilize permutation invariant ANNs with respect to a cell's neighbors, $\mathcal{N}_i(t_k)$ (see fig. 1 (C) for an illustration): Inspired by Ref. 114, this is achieved by partitioning a cell's ANN into (i) a sensory part, $f_\theta^{(s)}(\cdot)$, preprocessing the state of each neighboring cell separately (*i.e.*, its own, $\varepsilon_i(t_k) = f_\theta^{(s)}(\mathbf{c}_i(t_k))$, and of all neighboring states, $\varepsilon_{i_\nu}(t_k) = f_\theta^{(s)}(\mathbf{c}_{i_\nu}(t_k))$) into a respective sensor embedding, $\mathcal{E}(\mathcal{N}_i(t_k)) = (\varepsilon_i(t_k), \varepsilon_{i_1}(t_k), \dots, \varepsilon_{i_N}(t_k))$. These neighbor-wise sensor embeddings are (ii) aggregated by mean (along the neighbor dimension) into a context vector $\mathbf{s}_i(t_k) = \langle \mathcal{E}(\mathcal{N}_i(t_k)) \rangle_\mathcal{N} \in \mathbb{R}^s$ of fixed size s, which is then used as the input of (iii) a controller ANN, $f_\theta^{(c)}(\cdot)$, potentially with recurrent feedback connections, that eventually outputs the cell's action, $\mathbf{a}_i(t_k) = f_\theta^{(c)}(\mathbf{s}_i(t_k))$; for details we refer to appendix A.

Due to the mean-aggregation of a cell's sensory embedding, each cell completely loses its ability to spatially distinguish between neighboring (and even its own) state inputs and thus fully integrates into the tissue locally. We would like to stress the close relation of our approach to the concept of breaking down the computational boundaries of a cell's "*Self*" via *forgetting*[93] and to the scaling of goals from a single agent's to a system-level objective[95].

To model the developmental process of morphogenesis, we here employ NCAs on a two-dimensional $N_x \times N_y$ square grid with the objective that all cells of the grid assume their correct, predefined target cell type, $\hat{g}_i$, after a fixed number of N_D developmental time steps, starting from an initial cell state configuration $\mathbf{c}_i(0)$. We attribute a number of N_G elements $\mathbf{g}_i(t_k) \in \mathbb{R}^{N_\mathrm{G}}$ of the N_C-dimensional cell state $\mathbf{c}_i(t_k) \in \mathbb{R}^{N_\mathrm{C}}$ of an NCA as indicators for expressing one of $1, \ldots, N_\mathrm{G}$ discrete cell types, such that $\mathbf{c}_i(t_k) = \mathbf{g}_i(t_k) \cup \mathbf{h}_i(t_k)$; the remaining $N_\mathrm{H} = (N_\mathrm{C} - N_\mathrm{G})$ elements of the cell state represent hidden states $\mathbf{h}_i(t_k) \in \mathbb{R}^{N_\mathrm{H}}$ of a cell that can be utilized by the NCA for intercellular communication. We explicitly define each cell's type, $g_i(t_k)$, as the argument (*i.e.*, the index) of the maximum element of the N_G-dimensional indicator vector $\mathbf{g}_i(t_k)$:

$$g_i(t_k) = \underset{\mathbf{g} \in \mathbb{R}^{N_\mathrm{G}}}{\arg\max} \left(\mathbf{g}_i(t_k) \right). \tag{2}$$

Training an NCA to assemble a predefined target pattern (realized by a set of $N_j = N_x \times N_y$ target cell types $\{\hat{g}_1, \ldots, \hat{g}_{N_j}\}$ for the entire grid) thus boils down to finding a suitable set of NCA parameters (c.f., "Genotype" in fig. 1 (E)) that minimizes the deviation of each cell i's type $g_i(t_\mathrm{D})$ from $\hat{g}_i$ after t_D developmental time steps, *i.e.*, after the developmental stage of the virtual organism (c.f., "System-level Phenotype" in fig. 1 from left to right, and details below). Here, we are interested in the *evolutionary implications of biologically inspired multi-scale competency architectures*, the latter being modeled by our morphogenetic NCA implementation. We thus introduce in section II B, and utilize in section III, evolutionary algorithms to evolve suitable sets of NCA parameters that maximize a fitness score based on comparing the "final" cell types of the NCA, $g_i(t_\mathrm{D})$, with the predefined target cell types $\hat{g}_i$.

B. Neuroevolution of NCAs: an Evolutionary Algorithm approach to Morphogenesis

Evolutionary algorithms (EAs) are heuristic optimization algorithms that maintain and optimize a set, *i.e.*, a population, $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_{N_\mathrm{P}}\}$, of parameters, $\mathbf{x}_j \in \mathbb{R}^\mathrm{X}$, also termed individuals, over successive generations to maximize an objective function, or a fitness score, $r(\mathbf{x}_j) : \mathbb{R}^\mathrm{X} \to \mathbb{R}$. Inspired by the ideas of natural selection and the DNA-based reproduction machinery of biological life, EAs (i) predominantly select high-fitness individuals of a given population for reproduction, and utilize (ii) crossover and (iii) mutation operations to generate new offsprings by (ii) merging the genomic material of two high-quality individuals from the current population, $\mathbf{x}_o = \mathbf{x}_j \bigoplus \mathbf{x}_k$, and (iii) occasionally mutating the offspring genomes, $\mathbf{x}_o \to \mathbf{x}_o + \xi_\mathbf{x}$ by adding (typically Gaussian) noise to the parameters; the $\bigoplus$ symbol indicates a genuine merging operation of two genomes, which may depend on the particular EA implementation. In that way, a population $\mathcal{X}$ of individuals is guided towards high fitness regions in the parameter space $\mathbb{R}^\mathrm{X}$, typically over many generations of successive selection and reproduction cycles (i)-(iii).

In contrast to biological life, many use-cases of EAs do not require a distinction between individuals in the parameter space, *i.e.*, *genotypes* $\mathbf{x}_j$, and the corresponding organisms in their natural environment, *i.e.*, *phenotypes*, $\mathbf{p}_j$: while the genetic crossover and mutation operations of biological reproduction rely on bio-molecular mechanisms on the level of RNA and DNA, *i.e.*, are performed in the genotype space, selection typically happens at the much more abstract level of an organism's natural environment, *i.e.*, in the phenotype space. Carrying this through computationally can be resource-demanding, depending on the complexity of a simulated environment. Nevertheless, to address the asymmetry between genotypes and phenotypes in multi-scale competency architectures, it is essential to evaluate the EA's fitness score in the phenotype space instead of the genotype space, $r(\mathbf{x}_j) \rightarrow r(\mathbf{p}_j)$.

We explicitly separate the genotype and phenotype representation of individuals by introducing a biologically inspired *developmental layer*[1] in between genotypes and phenotypes, $\mathbf{x}_j \xrightarrow[Layer]{Dev.} \mathbf{p}_j$, as illustrated in fig. 1. More precisely, we follow section II A and model the developmental process of morphogenesis *in silico* by utilizing NCAs: We treat an NCA j's parameters, such as the set of $i = 1, \ldots, N_j$ initial cell states $\mathbf{x}_j^{(\mathrm{S})} = \{\mathbf{c}_i(0)\}_j$ and the corresponding ANN parameters $\mathbf{x}_j^{(\mathrm{F})} = \theta_j$, as the (virtual) organism's genome,

$$\mathbf{x}_j = \mathbf{x}_j^{(\mathrm{S})} \cup \mathbf{x}_j^{(\mathrm{F})} = (\{\mathbf{c}_i(0)\}_j, \theta_j)\,, \tag{3}$$

explicitly partitioning the genome into a structural (S) and a functional (F) part, as indicated by the superscripts. We then perform a fixed number of t_D developmental steps employing eq. (1), and interpret the corresponding set of "final" cell types $\{g_i(t_\mathrm{D})\}_j$ of the entire NCA, c.f., eq. (2), as the mature phenotype,

$$\mathbf{p}_j = \{g_i(t_\mathrm{D})\}_j, \tag{4}$$

representing a two-dimensional tissue of cells.

In an effort to evolve the parameters, $\mathbf{x}_j$, of an NCA j to achieve morphogenesis of a two-dimensional spatial pattern of cell types, $\mathbf{p}_j$, that resembles a pattern of predefined target cell types, $\{\hat{g}_1, \ldots, \hat{g}_{N_\mathrm{j}}\}$, of a total of N_j cells on an $N_x \times N_y$ square grid (see section II A), we define the phenotype-based fitness score $r(\mathbf{p}_j)$ as

$$r(\mathbf{p}_j) = (2n_j^{(\mathrm{G})} - N_j) + r_T\, n_j^{(\mathrm{T})} - r_S\, n_j^{(\mathrm{S})}, \tag{5}$$

where (i) $n_j^{(\mathrm{G})}$ is the number of correctly assumed cell types $g_i(t_\mathrm{D}) = \hat{g}_i$ after t_D developmental steps, (ii) $n_j^{(\mathrm{T})}$ is the number of time steps at which the entire target cell type pattern is correctly assumed, *i.e.*, whenever $g_i(t_k \leq t_\mathrm{D}) = \hat{g}_i$ for all i, and (iii) $n_j^{(\mathrm{S})}$ is the number of successive time steps, t_s and $t_{s+1} \leq t_\mathrm{D}$, where all cell types stagnate, *i.e.*, where $g_i(t_{s+1}) = g_i(t_s)$ for all i. With eq. (5), we thus reward the entire NCA j by counting all correctly assumed cell types after t_D developmental steps (while discounting all incorrect cell types $g_i(t_\mathrm{D}) \neq \hat{g}_i$), we reward maintaining the target pattern over time with a factor of r_T, and discount a stagnation of a suboptimal pattern over time by a factor of r_S. We consider the problem solved if a final fitness score of $N_j = N_x \times N_y$ is reached. Notably, there is no explicit fitness or reward feedback on the level of the uni-cellular agents in our system; the fitness score is solely used as the selection criterion for sampling the next

evolutionary generations, so the cellular collective needs to evolve an intrinsic signaling mechanism to successfully perform the requested morphogenesis task.

The here proposed setting of genotypes, $\mathbf{x}_j$, corresponding phenotypes, $\mathbf{p}_j$, and associated fitness scores, $r(\mathbf{p}j)$, given by eqs. (3) to (5), respectively, can be used in combination with any black-box evolutionary- or genetic algorithm. We rely on the well-established *Covariance Matrix Adaptation Evolutionary Strategy* (CMA-ES)[103] to simultaneously evolve the set of initial cell state configurations (*i.e.*, structural genes, $\mathbf{x}_j^{(\mathrm{S})}$) and the set of corresponding ANN parameters of an NCA (*i.e.*, functional genes, $\mathbf{x}_j^{(\mathrm{F})}$) with the objective of a purely self-orchestrated formation of a two-dimensional spatial tissue as illustrated by fig. 1 and described by eq. (1).

III. RESULTS

A. The System: An Agential Substrate Evolves to Self-Assemble the Czech Flag

Evolution works with an active rather than a passive substrate, *i.e.*, with biological cells with agendas of their own[1]. Thus, at every stage of development during morphogenesis collective decisions are made at vastly different length- and time scales within an organism, guiding the formation of the mature phenotype. We aim to model exactly this process via Neural Cellular Automata (NCAs) described in section II A and employ evolutionary algorithms (EAs) to evolve the parameters of such NCAs, so the latter perform well on a target morphogenesis task, see section II B. Without loss of generality, we consider an $N_x \times N_y = 8 \times 8$–Czech flag pattern (as a more complex version of the classic French Flag problem of morphogenesis[115,116]) as the target pattern for our *in silico* morphogenesis experiments, with a fixed number of $N_j = 64$ cells in total, $N_\mathrm{G} = 3$ distinct cell types (colored blue, white and red, respectively) and $N_\mathrm{H} = 1$ hidden state, which renders the dimension of the NCA's cell state as $N_\mathrm{C} = 4$. We use a square grid of cells with $N = 8$ neighbors per cell and with fixed boundary conditions (see appendix C for details).

Starting from a genotype $\mathbf{x}_j$ defined in eq. (3), we perform a number of $t_D = 25$ developmental steps per morphogenesis experiment to "grow" a phenotype $\mathbf{p}_j$, described by eq. (4), based on which the fitness score $r(\mathbf{p}_j)$ is evaluated following eq. (5) (see fig. 1 for an illustration of this process). During this entire process, we limit the magnitude of the numerical cell state values $\mathbf{c}_i(t_k)$ at all time steps t_k to the interval $l_\mathbf{c} = [-3, 3]$, and, analogously, limit the magnitude of the proposed actions $\mathbf{a}_i(t)$ of each uni-cellular agent to the interval $l_\mathbf{a} = [-1, 1]$. This is achieved by clipping the numerical values of $\mathbf{c}_i(t_{k+1})$ after a cell state update described by eq. (1), and the ANN outputs $\mathbf{a}_i(t)$ to the respective limits $l_\mathbf{c}$ and $l_\mathbf{a}$. The noise level ξ_c defined in eq. (1) is counted in units of the action limits, $\max(l_\mathbf{a})$, and is thus sampled from a Gaussian distribution with zero mean and standard deviation ξ_c independently for each of the $N_\mathrm{C} = 4$ cell state elements, thus affecting the cell state updates during development; the actual numerical values for the hyperparameters above turned out to be well suited for the problem at hand, especially to reasonably compare and discuss simulation results for the means of this contribution, but are not crucial for the more general aspects on the evolutionary implications of multi-scale intelligence discussed here.

To study the effects of different types of decision-making machinery within a cell, we utilize two different architectures for the NCA's artificial neural networks (ANNs), a

Feed Forward (FF) and a recurrent ANN inspired by gene regulatory networks[117–120] (RGRNs)[121]. Notably, the RGRN-agent architecture augments cells with an internal memory that is independent of their states in the NCA and can thus not be accessed by the cells' neighbors. To balance the length of the structural genome $\mathbf{x}^{(\mathrm{S})}$ and functional genome $\mathbf{x}^{(\mathrm{F})}$ defined in eq. (3), the two ANN architectures, FF and RGRN, are chosen such that the number of parameters $N_{\mathrm{FF}} = 192$ and $N_{\mathrm{RGRN}} = 164$ is roughly the same as the number of initial cell states $N_j \times N_C = 64 \times 4 = 256$. Thus, the ANNs utilized here - and detailed in table I of appendix A - are tiny compared to Ref. 101.

For each experiment of evolving an NCA's parameters, *i.e.*, for each independent run of the EA, we typically utilize a population, $\mathcal{X}$, of $N_{\mathrm{P}} = 96$ individuals and a maximum number of $N_{\mathrm{M}} = 2000$ generations. As the EAs ultimate fitness criterion, we consider the average $F_j = \langle r(\mathbf{p}_j) \rangle_{N_{\mathrm{E}}}$ of $N_{\mathrm{E}} = 8$ statistically independent fitness scores $r(\mathbf{p}_j)$ of corresponding morphogenesis simulations starting from an individual j's genotype $\mathbf{x}_j$ and resulting at a corresponding phenotype $\mathbf{p}_j$ after t_D developmental steps; the developmental program described via eq. (1) is imperfect due to the developmental noise applied to the cell state updates and can thus lead to different, noise-induced phenotypic realizations. Typical values used here for the corresponding reward factors defined in eq. (5) are $r_T = 0.25$ and $r_S = 0.5$. We consider the problem solved if a fitness of $F_j = \max(n_j^{(\mathrm{C})}) = N_j = 64$ is reached, but since we reward individuals to maintain the target pattern over time (via r_T), the maximum possible fitness score after t_{D} developmental time steps is $\max(r_j(\mathbf{p}_j)) = 70.25$ in this example. Further details about the hyper-parameters of the EA and afforded computational resources can be found in appendix D.

B. Direct vs. Multi-scale Encoding: Cellular Competencies affect System Level Evolvability

We aim in this contribution to investigate the evolutionary implications of biologically inspired multi-scale competency architectures[1,94]. Thus, we compare two qualitatively different evolutionary processes both with the objective of morphogenetic pattern formation but whose genomes either (i) directly encode phenotypic features of a two-dimensional target pattern (cf. fig. 1 (A)), or (ii) encode cellular competencies of a multi-scale architecture that gives rise to the same phenotypic features (cf. fig. 1 (C)). Notably, different definitions of direct and indirect encodings in multi-agent systems have been used in the literature[122]. Here, we specifically distinguish between structural parameters, $\mathbf{x}_j^{(\mathrm{S})} = \{\mathbf{c}_i(0)\}_j$, in the search space that directly encode features of the phenotype, *i.e.*, specific initial cell types, $g_i(0) \approx \hat{g}_i$, and functional parameters, $\mathbf{x}_j^{(\mathrm{F})} = \theta_j$, that indirectly, or rather functionally encode the target pattern by parametrizing the intercellular communication and intracellular information processing competencies of the NCA that facilitate the self-orchestrated pattern formation process.

If no ANN at all were present in our model, *i.e.*, $\theta_j = \{\}$, and in the absence of noise, $\xi_{\mathbf{c}} = 0$, we would re-establish a direct mapping between genotype and phenotype, as $\mathbf{c}_i(0) = \mathbf{c}_i(t_{\mathrm{D}})$, and thus a direct encoding of the target cell type pattern can be achieved, $g_i(0) = g_i(t_{\mathrm{D}})$. However, by default, we allow each cell to successively regulate its own cell state towards a target homeostatic value via an iterative perception-action cycle defined by eq. (1) and, moreover, to communicate in that way with neighboring cells. More specifically, each cell updates its cell state solely based on its own and the states of

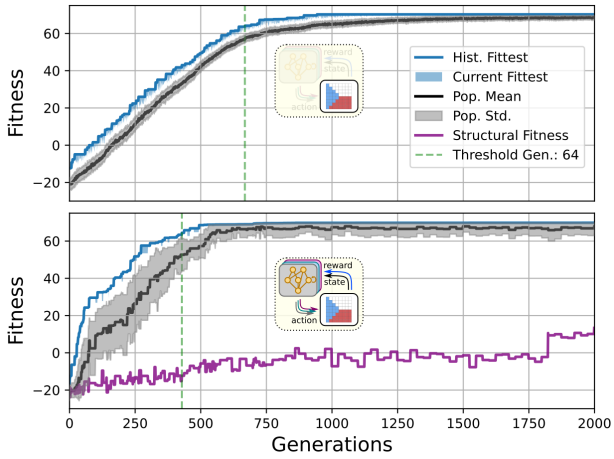

FIG. 2: Typical fitness trajectory over several generations of CMA-ES[103] of an NCA-based 8×8 Czech-flag morphogenesis task without (top) and with competency (bottom), corresponding to (i) direct and (ii) an multi-scale competency encoding of the target pattern as discussed in the text, representative for related experiments at similar system parameters (c.f., fig. 3). We present the historically- (blue) and currently-best fitness value per generation (light blue), the current structural fitness (purple), and the mean (black) and variance (gray) of the fitness of the entire population; in the top panel, the structural and phenotypical fitness are equivalent, thus only the latter is shown. The task is solved when a final fitness score of $F_j = 64$ is reached (marked by the green dashed line), *i.e.*, when $8 \times 8 = 64$ cell types are correctly assumed after $t_{\mathrm{D}} = 25$ developmental steps. The cartoon insets represent the perception-action cycle of the NCA, assembling an initial (random) arrangement of cell types into the target pattern; for the direct case (top panel), the NCA's ANN is disabled, which is illustrated by masking the agential parts in the cartoon.

its adjacent neighbors which, in turn, update their states based on their respective local environment. We explicitly avoid direct environmental feedback to the cells' perception (such as an individual or collective reward signal) but fully restrict the NCA to intercellular communication (via cell state updates) and intracellular information processing. These uni-cellular agents thus exhibit a certain level of problem-solving competencies that can be utilized for the challenge at hand, in our case for a collective system-level objective of forming a specific two-dimensional target pattern[95,101,102].

With the explicit partitioning of the genome into a structural part, *i.e.*, $\mathbf{x}_j^{(\mathrm{S})}$, and a functional part, *i.e.*, $\mathbf{x}_j^{(\mathrm{F})}$, we can study the effect of direct vs. multi-scale, or competency-driven encoding of phenotypic traits in the process of evolution, and, moreover, quantitatively tackle the question whether *competent parts affect the process of evolution and evolvability*. In any case, the initial cell state pattern is given by the structural part of the genome. Thus, in the absence of noise and without any active functional part in the genome, the set of initial cell states directly represents the final pattern, while otherwise cell states can either be modified passively by noise in the system or actively through actions by the cells during the developmental stage. Thus, we employ CMA-ES[103] to either evolve the (i) structural, or both (ii) the structural and functional part of the genome of an NCA simultaneously with the shared objective of self-assembling an 8×8 Czech-flag pattern in $t_{\mathrm{D}} = 25$ developmental time steps in the presence of noise, $\xi_{\mathbf{c}} = 0.25$ (cf., sections II B and III A for details). More explicitly, in case (i) we restrict the evolutionary process to search only the space of direct phenotypic encodings, while in case (ii) we allow evolution to evolve both a structural and a functional part of the genome, thus giving it the opportunity to prioritize one over the other (the results of this experiment are presented in fig. 2).

We can see in fig. 2 that both the evolution of the (i) direct and (ii) multi-scale encoding schemes of the target pattern can be achieved with the presented framework and a fitness threshold of $F_j = 64$ is reached after $\approx 300 - 600$ generations, thus solving the problem. However, depending on the encoding scheme (i) or (ii), we can identify clear qualitative differences in the strategy and the "efficiency" of the evolutionary process *i.e.*, how many generations it takes to reach a certain fitness threshold and eventually converge (c.f., top and bottom panel of fig. 2, respectively): The respective fitness score of the direct case (i) grows steadily and almost monotonically over successive generations until the threshold of $F_j = 64$ is reached after 668 generations for that particular run, and the EA converges at a maximum fitness of $\max F_j^{(i)} = 70.25$ after 942 generations (see section III A for details on the threshold fitness values). In contrast, the evolutionary process of the multi-scale case (ii) undergoes significant leaps as reflected by the corresponding fitness score which can increase rapidly if a suitable innovation, *i.e.*, a favorable crossover or mutation event in the functional parameters θ_j, occurs; the initial standard deviation of the fitness of the entire population is significantly larger compared to the direct case (i), yet the threshold fitness of $F_j = 64$ is reached in 428 generations, and the EA converges after 679 generations (although at a lower maximum fitness of $\max F_j^{(ii)} = 69$).

The results presented in section III B are based on selected evolutionary optimization runs, that are representative of related experiments with similar parameterizations. However, one should keep in mind that such results are always susceptible to chance in initial conditions or mutations in the EA, but also to the developmental noise; moreover, hyperparameters of the evolutionary search or even the specific ANN architectures can influence the evolvability of such NCA systems. Thus, we present in section III C below a more statistically significant analysis of the evolutionary implications of direct and multi-scale encodings under various conditions of the cellular agents' competency levels and the developmental noise.

Our separation of the genotype, $\mathbf{x}_j$, into a structural, $\mathbf{x}_j^{(\mathrm{S})} = \{\mathbf{c}_i(0)\}_j$, and into a functional part, $\mathbf{x}_j^{(\mathrm{F})} = \theta_j$, moreover allows us to extract the structural (or genotypic) fitness along an entire evolutionary history: We define the structural fitness as the fitness score $r(\mathbf{p}_j^*)$ of a phenotype $\mathbf{p}_j^*$ with evolved structural genes $\{\mathbf{c}_i(0)\}_j$ but with disabled agency $\theta_j \rightarrow \theta_j^* = \{\}$. Notably, in the direct case (i) we have $\mathbf{p}_j = \mathbf{p}_j^*$, which is illustrated in fig. 1 (A) and reflected in the top panel of fig. 2; the structural fitness of the multi-scale case (ii) is explicitly visualized in the bottom panel of fig. 2. In the latter case, the structural fitness remains essentially detached from the phenotypic fitness, $\mathbf{p}_j^* \approx 0 \ll \mathbf{p}_j$ during the entire evolutionary history (which also explains the convergence to a suboptimal maximal fitness level of $\max(F_j) = 69$ in this particular NCA solution, as the final Czech flag pattern first needs to be assembled from the corresponding imperfect initial cell configurations, $\mathbf{x}_j^{(\mathrm{S})}$). This all suggests that, in contrast to (i), the EA in (ii) can make the most use of exploring the functional part of the genome, *i.e.*, the space of behavior-shaping signaling and information processing[1], and, in turn, that the mere presence of competent parts drastically changes the search space accessible to evolution[3]; to illustrate this explicitly, we present in appendix E an illustration of the evolution of the morphogenesis process.

Interestingly, we still observe a slow but steady increase of the structural fitness in the long term in case (ii) owed to the small additional reward signal, r_T, reinforcing the cellular agents to maintain the target pattern over time. This can most efficiently be achieved if the agent starts from a perfect set of initial cell types, representing a particular sub-space

in the parameter space that might not necessarily be easily accessible to the EA at all stages during the evolutionary search. However, we would like to stress that such a slow transfer of problem-specific competencies from an agential, highly adaptive functional part, $\mathbf{x}_j^{(\mathrm{F})}$, to a rather rigid structural part, $\mathbf{x}_j^{(\mathrm{S})}$, of the genome could be a manifestation of the Baldwin effect[14]. Through a computational lens, such a competency transfer would also allow, as soon as the structural part of the genome is reliable enough, to re-purpose the system's competency to adapt to other, independent tasks, and thus may facilitate the in biology ubiquitous effect of polycomputing in related systems[123].

This all illustrates that an agential material[1,94], or more precisely a substrate composed of competent parts, can have significant effects on the process of evolution and evolvability, especially for morphogenesis tasks. We thus conclude that, if competent parts are available, evolution prefers exploiting competency over direct encoding - if the environment requires competency at all (see discussion in section III C). This leads to the conclusion, that "competency at the lowest level greatly affects evolution and evolvability at the system level."

C. Evolution exploits Competency over Direct Encoding, if necessary

Here, we investigate the effects of varying different levels of competency at the cellular level of a multi-scale competency architecture on the evolutionary process of morphogenesis. More specifically, we introduce the *decision-making probability* (I), P_{D}, which constrains the ability of each cell individually to perform cell state updates in the environment: P_{D} defines the probability at which a proposed cell state update of each individual cell in the NCA is executed (or otherwise omitted). Thus, varying the decision-making probability from $P_{\mathrm{D}} = 0$ to $P_{\mathrm{D}} = 1$ smoothly transitions the system's behavior from a direct encoding scheme without competency to an increasingly reliable multi-scale competency architecture (c.f., fig. 2).

Another, somewhat hidden level of competency we already discussed in section II A is each cell's ANN architecture: An RGRN-agent with internal memory can acquire and execute tasks differently than a simpler FF-agent without any feedback connections except for its cell state $\mathbf{c}_i(t_k)$. Comparing the evolutionary implications of such functionally different ANN architectures is, however, not trivial, and is thus kept to a minimum here.

However, we parameterize both FF and RGRN agents such that their controller part of the ANNs (c.f., fig. 1 (C), section II A and appendix A) are (II) stacks of R redundant copies of the same controller ANN, each copy with its own set of parameters, which take the same pre-processed aggregated sensor embedding as input, and whose individual outputs are averaged into a single action-output of a cell. Inspired by redundancy in error-correcting codes[105,106], we thus allow cells with higher values of this *redundancy numbers*, R, *i.e.*, with many alternative routes through the controller part of the ANN, to - in principle - integrate environmental signals more generally compared to $R = 1$, thus affecting the cells competency.

While scaling from $P_{\mathrm{D}} = 0$ to $P_{\mathrm{D}} = 1$ smoothly increases a cell's competency to reliably regulate its cell state, increasing R enhances the computational capacities of the unicellular agents. Henceforward, we interpreter (I) P_{D} and (II) R as two competency levels in our system which we can vary (I) continuously and (II) discretely.

Analogous to sections III A and III B, we thus utilize CMA-ES to evolve the genotypic parameters of an NCA to self-assemble the 8×8 Czech flag pattern under different condi-

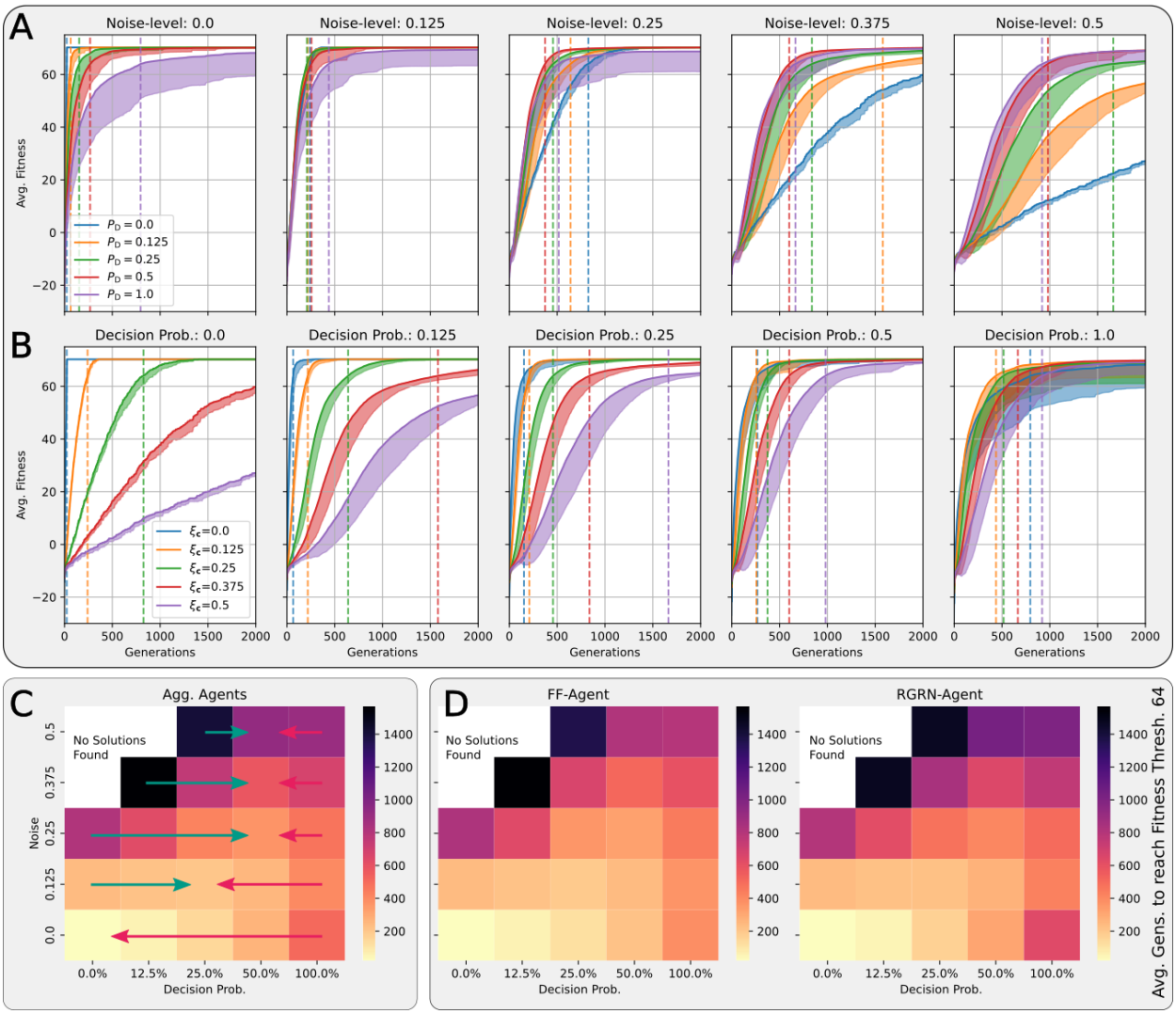

FIG. 3: **A** (**B**): The average fitness per generation of the best-performing individual in a population of 65 independent evolutionary processes of the 8x8 Czech flag task, evaluated from left to right at different noise levels (decision-making probabilities) and color-coded by the decision-making probabilities (noise-levels), respectively; solid lines mark average fitness values, the shaded area marks the standard deviation (to lower values only), and dashed lines indicate when an average fitness threshold of 64 is crossed, solving the problem. **C**: Heatmap of the average generation number when the fitness threshold of 64 is crossed at particular combinations of the decision-making probability and noise level as detailed in **A**, **B**; green and red arrows respectively indicate directions along P_{D} of increasing and decreasing values of the avg. fitness at fixed noise values. **D**: Same as **C** but partitioned by the respective FF-agent or RGRN-agent architectures used in the respective CMA-ES runs.

tions (I-II), and expose the cells to different noise-levels (III), $\xi_{\mathbf{c}}$, during cell state updates defined in eq. (1).

In fig. 3 (A,B) we present the corresponding fitness scores of a maximum of 2000 generations of CMA-ES for different noise levels $\xi_{\mathbf{c}} \in [0, 0.5]$, averaged over different values of the decision-making probability $P_{\mathrm{D}} \in \{0, 12.5\%, 25\%, 50\%, 100\%\}$ for both FF-agents and RGRN-agents. Moreover, for each realization of $\xi_{\mathbf{c}}$ and P_{D} we utilize experiments with different redundancy numbers $R \in \{1, 2, 4, 8, 16\}$ and employ 15 statistically independent EA runs for each parameter combination $\xi_{\mathbf{c}}$, P_{D} and R, and thus arrive at 75 statistically (and functionally, with respect to an agent's ANN architecture) independent fitness trajec-

tories per (P_{D}, $\xi_{\mathbf{c}}$)-combination; see section III A and appendix D for more details on the EA parameters. In fig. 3 (C) we present the average number of generations it takes to solve the problem (to reach a fitness threshold of $F_j = 64$) for each combination of P_{D} and $\xi_{\mathbf{c}}$, aggregated over the agents' ANN architectures, FF or RGRN, and the respective redundancy numbers R for 15 statistically independent EA-runs each; in fig. 3 (D) we present the data from fig. 3 (C) but separately for both ANN architectures.

We observe in fig. 3 that, depending on these two parameters, P_{D} and $\xi_{\mathbf{c}}$, for no- or very low noise levels, $\xi_{\mathbf{c}} \approx 0$, the evolutionary search is most efficient, *i.e.*, finds the solution in the least number of generations on average, for low values of the competency level $P_{\mathrm{D}} \approx 0$. Thus, in these situations, direct encoding (achieved via $P_{\mathrm{D}} = 0$) seems to be preferable to competency-driven encodings with $P_{\mathrm{D}} > 0$ (as indicated by the bottom red arrow in fig. 3 (C)); this is partly owed to the specific definition of the cell types $g_i(t_k)$ given by eq. (2), making a noise-less search very simple for the EA. However, for more realistic, noise conditions $\xi_{\mathbf{c}} > 0$, the situation changes drastically: With increasing noise level, the evolutionary efficiency of NCA's with higher competency levels is significantly larger compared to low competency levels and, especially, to the direct encoding scheme (as indicated by the green arrows in fig. 3 (C)); for noise levels of $\xi_{\mathbf{c}} = 0.375$ and 0.5, the EA does not even find solutions for the direct encoding case with $P_{\mathrm{D}} = 0$ in 2000 generations as cell state updates become increasingly necessary to counteract the noise in the system. There is a clear trend of increasing the evolutionary efficiency in our *in silico* morphogenesis experiments by increasing the competency level for increasingly difficult environments with high noise levels.

Thus, we conclude that scaling competency has a strong effect on the process of evolution, and in realistic situations (with moderate to high noise) competency may greatly improve the evolutionary efficiency and evolvability of collective self-regulative systems.

It might be noteworthy, that for evolving the 8×8 Czech flag pattern, essentially no qualitative difference in the evolutionary efficiency between FF-agents and RGRN-agents with the given number of parameters was observed. Also, the evolutionary implications of utilizing a number of $R > 1$ redundant copies within the controller ANNs of an NCA's cells is much less pronounced, compared to the results depicted in fig. 3, as can be seen in fig. 11 of appendix G. However, for more advanced problems such as assembling a 9×9 smiley-face pattern (see appendix F), RGRN-agents seems to outperform a simpler FF-agent significantly in terms of evolutionary efficiency. Moreover, a larger redundancy number of $R \geq 4$ is required by the evolutionary process to more efficiently evolve an NCA's functional parameters compared to a direct encoding scheme, hinting at a capacity bottleneck of the deployed ANNs.

D. There is a Trade-off between Competency and Direct Encoding depending on Developmental Noise.

A careful analysis of the results shown in fig. 3 reveals, that the largest competency level of $P_{\mathrm{D}} = 1$ does not result in the highest evolutionary efficiency for any presented noise level. On the contrary, populations with slightly lower competency levels of $P_{\mathrm{D}} = 0.5$ or even $P_{\mathrm{D}} = 0.25$ perform best at noise levels $\xi_{\mathbf{c}} \in \{0.25, 0.375, 0.5\}$ and 0.125, respectively (as indicated by the green and red arrow-ends in fig. 3 (C)). In fact, cells with an initially random genome (comprising the ANN and initial cell state parameters) that are forced to make "uninformed", *i.e.*, initially random, decisions at every time step can in-

terfere with the performance of the EA, as even initially perfect cell state configurations will be destroyed during such a randomized developmental stage. We suspect that this leads to corresponding delays in the evolutionary search compared to situations where populations can better rely on the structural part of the genome. Indeed, populations with "overconfident" actions can be trapped in local optima for many generations at all stages of the EA, which, in our system, may only be resolved by very specific but random mutations of the functional part of the genome (as we show later through fig. 4 in section III D). This is reflected in fig. 3 (A,B) by the large deviations in the average fitness trajectories for large P_{D} values.

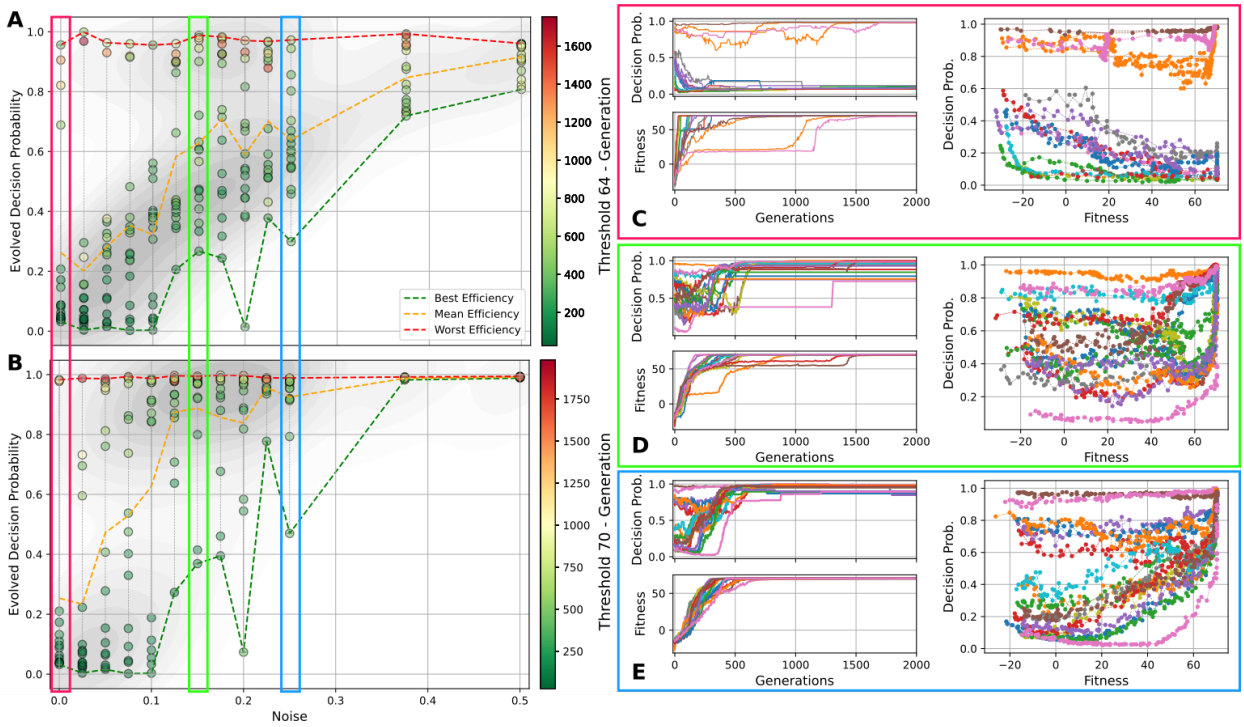

FIG. 4: **A**: The evolved decision-making probability P_{D} for different noise levels $\xi_{\mathbf{c}}$ when a fitness threshold of 64 for the 8×8 Czech flag task is reached; each symbol represents an independent lineage with a color-coding that indicates the number of generations it took for that particular lineage to cross the specified fitness threshold. The green/orange/red dashed lines indicate at which value of P_{D} the evolutionary process crossed the fitness threshold the fastest/on average/the slowest (*i.e.*, in the least, average, or largest number of generations) for each noise level. **B**: Same as **A** but with a fitness threshold 70. For both **A** and **B**, the red/green/blue frames emphasize the noise level $\xi_{\mathbf{c}} = 0, 0.125$ and 0.25 corresponding to panels **C**-**E**, respectively: The latter show the evolution of the decision-making probability/fitness (top/bottom left panel) and the value of the decision-making probability as a function of the corresponding fitness during the evolutionary process of each lineage (right panel) for all lineages at the specified noise level. Results are shown for an RGRN-agent architecture with redundancy $R = 1$, and are qualitatively similar to an FF-agent architecture.

The insights from above lead to the questions, of whether there is a "natural", or optimal competency level, with respect to the decision-making probability P_{D}, or whether a mutable competency level can be utilized by the evolutionary process to improve the efficiency of guiding a population towards high fitness regions in the parameter space. Thus, we include the decision-making probability as an additional *competency gene*, $\mathbf{x}_j^{(\mathrm{C})}$, into the NCA's genome, $\mathbf{x}_j \to \mathbf{x}_j = \mathbf{x}_j^{(\mathrm{S})} \cup \mathbf{x}_j^{(\mathrm{F})} \cup \mathbf{x}_j^{(\mathrm{C})}$, c.f., eq. (3), and we perform *in silico* morphogenesis evolution experiments of the 8×8 Czech flag pattern for different noise-levels, $\xi_{\mathbf{c}}$,

analogous to section III C. We analogously limit the numerical range of the competency gene $\mathbf{x}_j^{(\mathrm{C})}$ to the interval $[-3, 3]$, and extract the corresponding decision-making probability via $P_{\mathrm{D},j} = \frac{1}{2}(\tanh(\mathbf{x}_j^{(\mathrm{C})}) + 1)$. Notably, for the experiments shown in this sub-section, we use an L_2-regularization[124] on the genotypic parameters $\mathbf{x}_j = (x_{j,1}, \ldots, x_{j,N_x})$ through subtracting $r_{\mathrm{L}_2} \times \sum_{i=1}^{N_x} x_{j,i}^2$ from the fitness score defined in eq. (5), with $r_{\mathrm{L}_2} = 0.01$.

In fig. 4 (A,B) we present the evolved competency level for different noise levels after a fitness threshold of 64 and 70 is crossed, respectively, for 10 independent lineages per noise level for an RGRN-architecture. The problem is considered solved at a fitness of 64, but since we reward the NCAs to maintain the target pattern over time via $r^{(\mathrm{T})}$ in eq. (5), a higher maximal fitness score of 70.25 can be reached after t_{D} developmental steps for sufficiently long evolution. Thus, we here relate fig. 4 (A) to the evolutionary stage of having achieved the process of morphogenesis, and fig. 4 (B) of having achieved morphostasis. For both cases, we essentially see two strategies emerging (see also fig. 4 (C-E)): (i) one, where competency is maximized very early during the evolutionary process that then remains near the maximally possible value of $P_{\mathrm{D}} = 1$, and (ii) a hybrid strategy where a significantly lower competency level is assumed that still allows to solve the problem.

Notably, strategy (i) is predominantly pursued at high noise levels where large cell state fluctuations in the environment favor informed actions by the cellular agents. In contrast, the second strategy (ii) emerges more frequently in lineages evolved at low noise levels where, especially at very low noise levels $\xi_{\mathbf{c}} \approx 0$, most of the evolutionary processes result in solutions that avoid competency altogether and a direct encoding scheme (P_{D}=0) is evolved. Intermediate competency levels evolve in the corresponding intermediate noise regime. Following the trend of evolving morphogenesis (by crossing a fitness score of 64) to morphostatsis (by converging to the maximal fitness value of ≈ 70) in fig. 4 (A) through (B), we see that the two strategies, (i) and (ii), "sharpen" during the course of the evolutionary process, such that P_{D} predominantly converges to the minimally or maximally possible values of 0 and 1, depending on the environmental conditions.

We also illustrate the evolved competency level of the particular lineage at all noise levels in fig. 4 (A,B) at which the respective fitness threshold is crossed in the least-, and maximum number of generations (and on average) amongst all 10 independent lineages per noise level. This clearly reveals that evolutionary processes that follow a more direct encoding strategy (ii) can evolve the problem at hand efficiently - if this is permitted by the developmental noise. However, when increasing the noise level, the evolutionary process can afford to evolve - or put differently, increasingly relies on evolving - the multi-cellular intelligence of the NCA to perform morphogenesis and morphostasis, thus following a third strategy (iii) that integrates both strategies (i) and (ii) in a non-trivial way. We observe in fig. 4 (A) that the most efficient strategy for evolving morphogenesis seems indeed to be such a hybrid approach (iii), where a minimally necessary competency level is utilized at a specific noise level such that the corresponding evolutionary process can, again, be very efficient in solving the task.

Moreover, this also holds for the stage where morphostasis is reached, c.f., fig. 4 (B): Lineages that efficiently evolved to solve morphogenesis in our experiments also (typically) evolve to solve morphostasis efficiently. To emphasize this, we present in fig. 4 (C-E) the "temporal dynamics" of the population-wise highest fitness and the corresponding competency level per generation for all lineages at selected noise levels $\xi_{\mathbf{c}} =$

$\{0, 0.125, 0.25\}$; we also present for all corresponding lineages that have been evolved at these selected noise levels the genotypic competency level $P_{\mathrm{D,j}}$ against the corresponding phenotypic fitness scores r_j, and we find an apparent yet non-trivial relation between these two quantities: typically, an initial rise in fitness r_j in early generations is associated with a decline in $P_{\mathrm{D,j}}$ which is more pronounced at lower noise levels. For intermediate noise levels $0 < \xi_{\mathbf{c}} \ll 1$ we find that $P_{\mathrm{D,j}}$ often assumes a minimum (*i.e.*, a minimally required yet finite competency level) when the evolutionary process reaches a fitness level of ≈ 64. We suspect, that this allows the evolving morphogenetic process to establish good starting configurations based on changes in the structural genome, which can most efficiently be done at a minimal(ly necessary) competency level given a certain developmental noise level in the environment. However, the competency is then quickly pulled towards a maximum level of $P_{\mathrm{D,j}} = 1$ when the EA converges at a maximum fitness score of ≈ 70, at the morphostasis stage. For large noise levels, e.g., $\xi_{\mathbf{c}} = 0.25$ as depicted in section III D(E), the competency level rises with the corresponding fitness score in a much more monotonic way, emphasizing the necessity of the corresponding NCAs to utilize the cellular competency to solve the problem already at an early stage of the evolutionary process.

Curiously, we also see lineages that settle at the highest possible competency levels throughout their evolutionary history, even in conditions without noise, as can be seen in section III D(C): Here, an initial "frozen accident" may cause an entire lineage to maintain high competency levels due to a lack of diversity in the corresponding gene, although this is not even necessary to solve the task. However, these high competency levels early on during the evolutionary process can cause the population to stagnate at sub-optimal regions in the parameter space for many generations if the corresponding policy of the cells is sub-optimal but rigid to strategy changes via small mutations in the genome. The population seems "trapped", until a favorable mutation or crossover event occurs in the functional part of the genome of an individual, that guides the entire population towards higher fitness scores, eventually solving the problem. We suspect that this is also the reason for the lower evolutionary efficiency of the "most competent" configurations (with $P_{\mathrm{D}} = 1$) compared to slightly less competent cases (with $P_{\mathrm{D}} = 0.5$) of the experiments depicted in fig. 3[125].

Thus we conclude, that if the evolutionary process can afford to evolve its own competency level, there seems to be a trade-off - during the entire course of the evolutionary process - between "going direct" or "going competent", depending on the developmental noise. Moreover, randomly initialized starting conditions may favor either direct or multi-scale encoding strategies, which may not only affect the "final" competency level the evolutionary process converges to, but can also greatly influence the efficiency of the evolutionary process itself. In general, the most efficient strategy for evolving morphogenesis tasks seems to be a non-trivial tradeoff between finding a suitable initial cell state configuration that then allows the competency-based self-assembly of the target pattern to "kick in" and solve the task efficiently.

E. Competency can Lead to Generalization

We are ultimately interested in the question of whether a substrate of competent parts shows abilities to generalize to environmental conditions that have never been experienced by its evolutionary predecessors, and hence would allow the evolutionary process

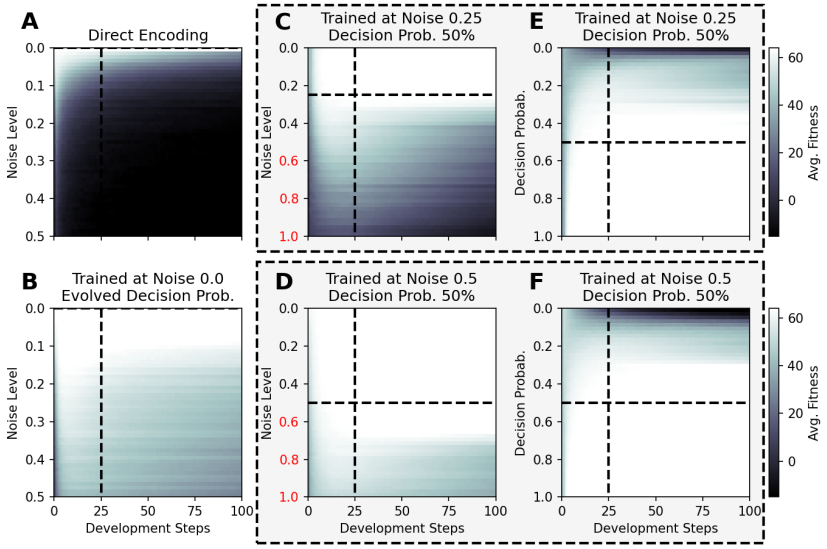

FIG. 5: The average fitness score of 100 independent evaluations of selected NCA results utilized at noise (**A-D**) and competency-level conditions (**E, F**) which have not been experienced during training for an increased total lifetime of 100 time steps. The respective NCAs have been evolved at zero-noise without competency (**A**), with evolvable competency (**B**), and under different noise conditions and decision-making probabilities (**C-F**), with a fixed number of $t_{\rm D} = 25$ developmental steps; results of all panels except for **B** are based on RGRN-agent architectures with training conditions given by titles and dashed lines. The data presented in panels (**C, E**) and (**D, F**) are respectively based on the same NCA solution (indicated by the dashed frames), while the noise level is varied in (**C, D**) at a fixed competency level of $P_{\rm D} = 0.5$ and the competency-level is varied in (**E, F**) at a fixed noise-level of ($\xi_{\mathbf{c}} = 0.25$, $\xi_{\mathbf{c}} = 0.5$)], respectively.

to adapt an organism to changing environmental conditions more efficiently compared to a direct encoding scheme. Thus, we systematically vary in fig. 5 the system parameters, *i.e.*, the noise level and the decision-making probability competency level, for selected NCA solutions of the Czech flag problem that have been trained with certain sets of the system parameters above.

For instance, we utilize NCA solutions that have been evolved to solve the 8×8 Czech flag problem in $t_{\rm D} = 25$ developmental steps (see above) under zero-noise conditions without and with evolvable competency. Here, we utilize such solutions for larger noise levels of $\xi_{\mathbf{c}} \in [0, 0.5]$ and for lifetimes of 100 time-steps and present the average fitness values of 100 statistically independent simulations at each particular noise level in fig. 5 (A, B), respectively - without any further evolutionary optimization. Analogously, we expose NCA solutions that have evolved with a competency level of $P_{\rm D} = 0.5$ and noise levels of $\xi_{\mathbf{c}} = 0.25$ and 0.5, respectively, to vastly different noise levels of $\xi_{\mathbf{c}} \in [0, 1]$ compared to the conditions during their respective evolutionary processes, and present the results in fig. 5 (C-D). Eventually, we again deploy the latter NCA solutions but vary the competency level $P_{\rm C} \in [0, 1]$ instead, at respectively fixed noise levels of $\xi_{\mathbf{c}} = 0.25$ and 0.5, with results depicted in fig. 5 (E-F). Notably, we only consider the "correctness" part of the fitness score, *i.e.*, the first term in eq. (5) by setting $r_T = 0$ and $r_S = 0$.

The results in fig. 5 demonstrate that the performance of the here evolved NCAs, optimized with evolutionary methods to assemble and maintain a target morphology over time at particular system parameters, differs greatly between NCA solutions that follow the direct- or multi-scale encoding paradigms when subjected to novel environmental conditions: The typical fitness over the lifetime of an NCA without competency that encodes the target phenotype pattern directly (c.f., fig. 5 (A)) is constantly affected by random fluc-

tuations and thus decreases in fewer time steps with increasing noise levels in a diffusive process; the duration of how long the corresponding maximum fitness score of 64 can be maintained, and the speed at which the fitness eventually decays during the lifetime of the here discussed 8×8 Czech flag NCA depends on the particular noise-level and on the values of the initial cell states, which are limited numerically to the interval $[-3, 3]$ for each cell. In contrast, NCA solutions with larger competency levels that have been evolved at finite noise-level conditions still perform well - and can maintain the target pattern for exceptionally long times - also when changing the system parameters dramatically, (c.f., fig. 5 (B-F)); note the noise-level axis of $\xi_{\mathbf{c}} = 0$ to 1, compared to maximum noise-levels of $\xi_{\mathbf{c}} = 0.5$ during training.

The results in panel fig. 5 (B) are especially curious, as the corresponding NCA has been trained to evolve its decision-making probability alongside the structural and functional parts of the genome at zero noise conditions. While no competency at all would have been required to solve this task, the presented NCA solution evolved to afford a maximum competency of $P_{\mathrm{D}} = 1$ (c.f., fig. 4 (C)). Strikingly, this particular NCA is capable of resisting much larger noise levels of $\xi_{\mathbf{c}} \approx 0.25$ while maintaining the pattern perfectly for at least $t_{\mathrm{D}} = 25$ steps, and the average fitness score of 100 independent solutions does still not drop below a certain threshold of $\approx 40 - 50$ for even higher noise levels and for 100 time steps. Notably, there appears to be a bifurcation of the long-term behavior of these NCA solutions (not shown here) where the NCA - in some realizations - maintains the target pattern perfectly for long times, while in other independent runs, the fitness drops quickly.

In this sub-section, we thus show that NCAs that have evolved to assembly and maintain a target pattern within a relatively short developmental stage are capable of maintaining the corresponding target pattern over much longer time scales - without any further optimization - and thus show great signs of functional, morphostatic generalizability. Moreover, the here-discussed *in silico* morphogenesis and morphostasis model systems are capable of handling - essentially on the fly - system-parameter combinations neither they, nor their evolutionary ancestors ever experienced before. Thus, we conclude that such multi-scale competency architectures[1], whose substrate is composed of competent rather than passive parts, can be more than capable of generalizing to changes in their environment - within reasonable boundaries, of course - by allocating robust problem-solving competencies at many scales[93,94].

F. Competency can Augment Transferability to New Problems

Deducing from the discussion in section III E about the generalizability of multi-scale competency architectures[1] towards changing environmental conditions, such systems should also exhibit increased evolvability and transferability properties to new problems: if such multi-scale competency architectures are capable of adapting their behavior towards changing environmental conditions on the fly during a single lifetime (c.f., fig. 5), this has great consequences for the evolutionary process when environmental conditions change.

Thus, we utilized the NCA solution discussed in fig. 4 (C) and fig. 5 (D) and performed subsequent CMA-ES on the 8×8 Czech flag problem at changed environmental conditions, *i.e.*, at higher noise levels: only a single or at most a handful of generations are necessary for solving the task even at intermediate and high noise levels of $\xi_{\mathbf{c}} = 0.25$ and 0.5 (not shown here).

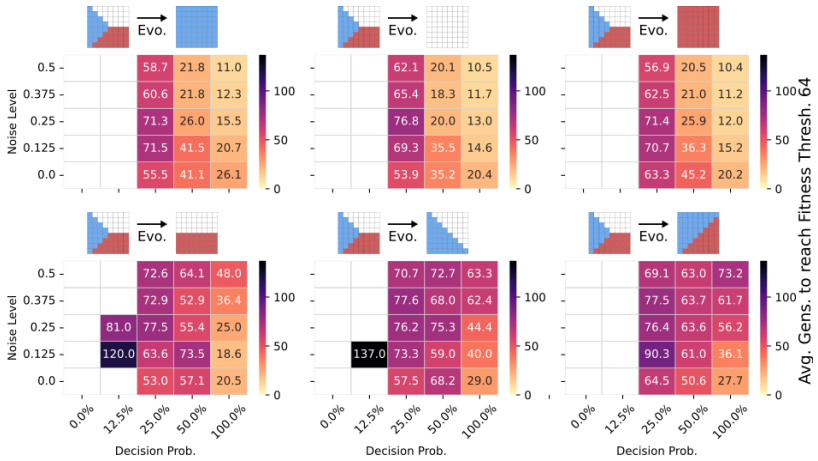

FIG. 6: The average number of generations it takes for the CMA-ES to adapt a pre-evolved NCA solution that can solve the 8×8 Czech-flag morphogenesis task to adapt, respectively, to the 8×8 blue-, white-, red-, and Viennese-, blue\white-, and blue/red-flag morphogenesis tasks instead (c.f., panel insets) and reach a correctness fitness score of 64. We specifically adapted Czech-flag NCA solutions that have been pre-evolved at a noise level of $\xi_c = 0.25$, but with corresponding competency levels according to the horizontal axis in fig. 3, and deploy CMA-ES for 1000 generations at the corresponding noise/competency-levels depicted here on the vertical/horizontal axis, and average over multiple CMA-ES runs and corresponding redundancy numbers, $R = 1, 2, 4, 8, 16$.

To emphasize the potential of transferability of multi-scale competency architectures, we here investigate the adaptation-capability of pre-evolved NCAs when their objective function is suddenly changed, *i.e.*, when the environment starts selecting for different target patterns than the one they have originally been evolved for. More specifically, we utilize NCA solutions from section III C, and discussed through fig. 3, which successfully solve the 8×8 Czech-flag task, and additionally perform 1000 evolutionary cycles of CMA-ES on a related 8×8 blue-, white-, red-, and Viennese-, blue\white, and blue/red-flag morphogenesis task for various noise and competency levels. We allow changes to both the structural and functional parts of the genomes of the pre-evolved NCA.

In fig. 6, we present the corresponding number of generations it takes for $10-60$ CMA-ES runs on average to adapt a pre-evolved, *i.e.*, "informed", NCA solution that can solve the 8×8 Czech-flag morphogenesis task to then solve the respective new morphogenesis task under different environmental conditions. We see a clear advantage in terms of evolvability and adaptability of pre-evolved individuals at high-competency levels (in contrast to individuals with lower competency levels) so that adaptation can happen in as few as ≈ 10 generations. While the Czech$\rightarrow$blue-, white- , and red-flag tasks are rather trivial (see top panels in fig. 6), computationally, the Czech$\rightarrow$Viennese-, blue\white, and blue/red-flag adaptation tasks (bottom panels in fig. 6) are more complicated. Still, the latter can be solved in as few as ≈ 20 generations compared to $\gg 100$ generations of evolving a corresponding randomly initialized NCA to solve the Czech-flag problem from scratch, as shown in sections III B and III C.

Thus we conclude, that pre-evolved (or "informed") competency at subordinate scales of a multi-scale competency architecture greatly enhances a collective system's capability of adaptation. Thus, a competent and informed substrate has great effects on a multi-scale competency architecture's evolvability towards changing environmental conditions and on the transferability of already acquired (evolved) solutions to new problems.

IV. CONCLUSION

We have investigated the evolutionary implications of multi-scale intelligence on the example of *in silico* morphogenesis of a two-dimensional tissue of locally interacting cells that are equipped with tuneable decision-making machinery. More specifically, we have utilized evolutionary algorithms (EAs)[103] to evolve the parameters of Neural Cellular Automata (NCAs)[100] on morphogenesis tasks under various conditions of the competency level of the uni-cellular agents and the developmental noise in the system.

In this model of a multi-scale competency architecture[1], a two-dimensional grid of locally interacting cells is tasked to self-assemble and maintain a global spatial target pattern of predefined cell types, here primarily of a two-dimensional, 8×8 Czech flag pattern[126]. Each uni-cellular agent's internal decision-making machinery is modeled by an artificial neural network (ANN), allowing these cells to independently perceive the cell states of their adjacent neighbors on the grid and propose actions to regulate their own cell state over time, thereby communicating with their neighbors. Both the ANN parameters and the initial cell states of all permanent cells represent the parameters of the NCA and are optimized by EAs for a specific *in silico* morphogenesis task at hand, thus forming the functional and structural part of the system's genome, respectively.

To investigate the effects of competency in a multi-scale competency architecture on the underlying evolutionary process, we vary (I) the reliability an NCA's uni-cellular agents can independently regulate their cell types during a noisy developmental stage. We thus specifically define a "competency level" parameter in our system as the *decision-making probability* at which proposed actions of uni-cellular agents are considered in the NCA's corresponding cell state updates (or omitted otherwise). This allows us to continuously scale the NCA's competency level from a direct encoding scheme of the target pattern (no competency) to a multi-scale competency architecture that self-assembles the pattern with perfect reliability in cell decision executions. Furthermore, we introduce (II) a variable number of redundant sub-modules in the NCA's ANN, each with an independent set of functional parameters, which we can control in our system as another "axis" of competency based on redundancy and computational capacity of the cells' decision-making machinery.

In large-scale simulations, we systematically vary these two competency levels (I, II), expose the corresponding NCA to different noise conditions (III), and perform several statistically independent evolutionary searches at each parameter combination (I-III). In that way, we demonstrate that an evolutionary process proceeds significantly more rapidly (on average) on noisy pattern formation tasks when evolving the parameters of a multi-scale competency architecture compared to evolving the target pattern directly (with no competency involved).

Our multi-scale competency architecture model and the corresponding evolutionary optimization process comprise several scales: At the smallest scale (1), each structural and functional gene is represented by a floating point number. The functional genes parameterize the behavior of artificial neurons (2), our atomic decision-making centers, which are then hierarchically arranged into layers of artificial neurons (3), sub-modules of interconnected layers (4), to an ANN with a predefined architecture (5). Thus, even the uni-cellular phenotypes (6) in our system - ANN-based agents that maintain a particular internal cell state - are composites of smaller (proto-competent) decision-making centers down the hierarchical ladder. The composite uni-cellular agents perceive the cell states

of their grid neighbors (7) on the NCA, perform potentially several cycles of internal calculations, and eventually update their own cell state in a single developmental step. In that way, clusters of different tissue types (8) may be formed in successive developmental steps. A fixed number of developmental steps comprise the lifetime of a single NCA, giving rise to a self-assembled phenotypic tissue of cell types on the entire grid of the NCA (9), *e.g.*, as in our case, to the Czech flag pattern. The quality of each individual in an evolutionary population of NCAs (10) is evaluated via a phenotypic fitness score, quantifying the deviation of the assumed cell types from a target pattern. Based on the fitness scores of a particular generation of NCAs, the genotypes of potentially better-adapted successor generations are successively sampled by the EA, closing the loop (1) and forming the largest scale in our system, an evolutionary lineage (11). Eventually, on a meta-scale (12), we compare the efficiency of the evolutionary process at different system parameters (I-III), *i.e.*, at different competency- and noise levels, by analyzing the fitness trajectories of statistically independent lineages evaluated at the same system parameters.

We demonstrate that especially in the presence of developmental noise, affecting cell state updates during morphogenesis, the evolutionary process favors a multi-scale competency-based realization over a direct encoding scheme of the target pattern. Moreover, when the competency level itself was left as an evolvable parameter to the EA, there appeared to be a non-trivial dynamical tradeoff in the evolutionary process' efficiency between exploiting the competency level of its components or the direct, prepatterning-like encoding of the target pattern. We thus report that under realistic conditions (*i.e.*, at moderate noise levels), an evolutionary process can be significantly more efficient when working with an agential- rather than a passive material[1,94].

Notably, we explicitly omit a reward or fitness feedback from the environment to the NCAs' uni-cellular agents' perception, restricting the cells' decision-making solely to local communication of cell state updates between grid neighbors. Thus, the cells need to figure out their own communication protocol such that their single-agent decisions align with the global (multi-agent) system-level objectives of assembling the correct target pattern. These uni-cellular competencies are acquired over evolutionary time scales and can be understood as emergent behavior-shaping signaling[1].

On a more technical note, we specifically employ permutation invariant ANNs as trainable update functions of the NCAs and successfully evolve the corresponding models to perform the here studied pattern formation tasks. We thus show that, contrary to previous assumptions[101,127], a perfect spatial resolution of neighboring cell states in an NCA is not necessary but that a mean-aggregated neighboring cell state can be sufficient for single cells to reliably contribute to the objective of a larger scale collective. Strikingly, we show that such uni-cellular agents do not even need to distinguish between their own states and the states of their neighbors to achieve this task, thus fully integrating into the tissue locally and essentially losing their individuality[93,95].

Also in contrast to Ref. 101 and similar work, we do not start our morphogenesis experiments from a single "alive" cell but instead evolve the initial cell states of all permanent cells on the grid of an NCA, while the uni-cellular agents are constantly challenged to correct their state from developmental noise (notably, a process reminiscent to the denoising steps of Diffusion Models[128–132]). This allows us to explicitly distinguish between the evolutionary implications of (i) direct and (ii) multi-scale competency-based encodings of the target pattern, where we either constrain the evolutionary process to (i) only

evolve the structural part of the genome, or to (ii) evolve both the structural and functional part simultaneously. Admittedly, the choice of the structural part of the genome limits the scalability of the approach, as the size of the structural genome will grow correspondingly with the number of cells in the system. However, as occurs with biomechanical[133], biochemical[134,135] and bioelectric prepatterning[8,94,136], the initial states of an NCA of moderate size could be seen as a coarse-grained scaffold, based on which an NCA of potentially much higher resolution can run its multi-scale competency-based developmental program to self-assemble a high-resolution target pattern[137]. Alternatively, we suggest utilizing a Compositional Pattern Producing Network (CPPN)[120,138] to indirectly encode the initial states of all cells on the grid of an NCA, allowing such a hybrid approach to perform *in-silico* morphogenesis at scale. Unfortunately, it has been proven difficult, if not unfeasible, to exactly reproduce predefined target patterns reliably with neuroevolution of CPPNs alone[139], which is why we here refrained from this approach; we emphasize, however, that gradient-based methods such as Neural Radiance Fields (NeRF)[140] to train CPPN-like architectures might be an interesting workaround.

We find that fully evolved NCA solutions, capable of performing the morphogenesis tasks discussed above, show great signs of generalizability toward changing the system parameters, and can - without any further evolutionary optimization or training - handle noise and competency levels that are vastly different from the training conditions. Consequently, this leads to increased evolvability of such competency-based models to changing environmental conditions: a subsequent evolutionary process can adapt a pre-evolved solution to altered environmental conditions within a handful or sometimes even a single generation. Moreover, we demonstrate that such pre-evolved NCA solutions can even quickly adapt to new, yet related problems. Specifically, we modified the objective function of our evolutionary process from the 8×8 Czech flag task to self-assemble a blue-, red-, white-, Viennese-, diagonal blue\white and blue/red flag instead, respectively. In most of these situations, an adaptation of an existing NCA solution to the new problem can be done in significantly fewer generations than evolving the initial 8×8 Czech flag task from a randomly initialized configuration. Typically, these adaptations happen the faster the larger the competency level of the NCA, while for the direct encoding scheme (or in situations with low competency) the structural part of the genome is too dominant to allow quick adaptations by the EA. This suggests that multi-scale competency architectures allow the underlying evolutionary process to not over-train on priors, thus augmenting adaptability through a competent substrate.

We conclude that not only can evolutionary processes efficiently utilize and bring forth the intriguing multi-scale problem-solving machines of biological life, but that the efficiency of such evolutionary processes, as well as the generalization abilities, evolvability, and transferability of the corresponding phenotypic outcomes, are strongly affected by the level of competency of the underlying agential material. An intriguing open question is whether this implies a positive feedback loop that enhances that quality over time. Judging from the considerable effects of scaling the competency in the here studied still shallow multi-scale system on a rather simple *in silico* evolutionary process (*i.e.*, CMA-ES[103]), it becomes increasingly evident that the vastly more complex multi-scale competency architecture of biological life cycles back and thus affects the process of evolution itself.

For future directions, our multi-scale competency framework is easily extendable to simulate tissue growth via cell migration or division actions proposed by the NCA's under-

lying ANNs. More specifically, our framework allows for a minimal set of biologically relevant uni-cellular actions, such as a cell state update, cell division, migration, cell death, and an identity operation, only constrained by the NCA's spatial grid. Furthermore, the framework is capable of handling flexible ANN architectures, potentially allowing us to investigate intriguing competencies such as active inference[141] through utilizing world model architectures[142] in a (neuro)evolutionary context. Our system, so far, has a fixed hierarchical architecture that deviates from the scale-free competency architecture of biological life with open-ended functional adaptation (where any abstraction layer becomes the basis for the next one). Thus, in future work, we aim to model precisely this behavior by introducing multiple layers of horizontal communication pathways in an NCA that the ANN-based agents can dynamically traverse in the vertical direction. Moreover, by choosing a proper fitness function related to measuring scale-invariant pattern formation[107,143], critical dynamics[144–149], or applying the free-energy principle[141,150], we are confident to achieve a biologically more accurate model of the scale-free dynamics and open-ended evolution of life. Such computational models could thus further quantitative studies of the communication strategies and boundaries of individual- and groups of cells in an agential, potentially adversarial Umwelt, with possible applications in individual- and collective aging (as morphostasis defects)[151,152] or cancer research[93,94,136].

ACKNOWLEDGEMENTS

We thank Hananel Hazan, Léo Pio-Lopez, Yanbo Zhang, Axel De Baat, Patrick Erickson, and Adam J. Goldstein for helpful discussions and their input. BH acknowledges an APART-MINT stipend from the Austrian Academy of Sciences. SR was supported by the European Union (ERC, GROW-AI, 101045094). M.L. gratefully acknowledges support via grant 62212 from the John Templeton Foundation, and a Sponsored Research Agreement from Astonishing Labs. The computational results presented have been achieved (in part) using the Vienna Scientific Cluster.

AUTHOR DECLARATIONS SECTION

The authors have no conflicts to disclose.

DATA AVAILABILITY STATEMENT

Computational protocols and numerical data that support the findings of this study are shown in this article, and in the Appendix.

Appendix A: Artificial Neural Networks

Inspired by biological neural circuits, an Artificial Neural Network (ANN) is an interconnected network of artificial neurons (AN)[153–156]. Each such AN maps a set of inputs, $\mathbf{x} \in \mathbb{R}^n$, onto a single number, $y \in \mathbb{R}$, usually through a non-linear filter, $\sigma(.)$: The output of an AN can be defined as a parameterized function, $y = \sigma(\mathbf{w} \cdot \mathbf{x} + b)$, with weights $\mathbf{w} \in \mathbb{R}^n$ and bias $b \in \mathbb{R}$[157].

Commonly organized in layers of ANs, a *Feed Forward* ANN represents a parameterized non-linear function, $\mathbf{y}^{(\text{out})} = f_\theta^{(\text{FF})}(\mathbf{x}^{(1)})$, transforming an input, $\mathbf{x}^{(1)} \in \mathbb{R}^{\mathrm{N}_0}$, over $i = 1, \dots, N_\mathrm{L}$ consecutive hidden layers of ANs, $\mathbf{y}^{(i)} \in \mathbb{R}^{\mathrm{N}_\mathrm{i}}$, to an output vector, $\mathbf{y}^{(\text{out})} \in \mathbb{R}^{N_\mathrm{L}}$. More specifically, the output $\mathbf{y}^{(i)} \in \mathbb{R}^{\mathrm{N}_\mathrm{i}}$ of layer i, defined by

$$\mathbf{y}^{(i)} = \sigma\left(\mathcal{W}^{(i)} \cdot \mathbf{x}^{(i)} + \mathbf{b}^{(i)}\right) \tag{A1}$$

becomes the input, $\mathbf{x}^{(i+1)} = \mathbf{y}^{(i)}$, to the next *deeper* layer, $i+1$, through layer-wise filtered dot-products with the weight matrices $\mathcal{W}^{(i)} = \{w_{jk}^{(i)}\} \in \mathbb{R}^{\mathrm{N}_\mathrm{i} \times \mathrm{N}_{\mathrm{i}-1}}$ and bias vectors $\mathbf{b}^{(i)} = (b_1^{(i)}, \dots, b_{\mathrm{N}_\mathrm{i}}^{(i)}) \in \mathbb{R}^{\mathrm{N}_\mathrm{i}}$.

Training an ANN thus boils down to optimizing a set of parameters, $\theta = \{w_{jk}^{(i)}, b_k^{(i)}\}$, i.e., the entire network's weights and biases, such that an input is mapped (with minimal deviation) to a desired output[113,158–160]. In this manuscript, we utilize ANNs as the trainable update function of a neural cellular automaton (NCA)[100] and optimize the corresponding ANN parameters via evolutionary algorithms to study the evolutionary implications of multi-scale intelligence on the example of morphogenesis.

Notably, in contrast to previous contributions of NCA-based morphogenesis[101], we do not rely on predefined convolutional filters in our ANN architectures to preprocess a cell's local environment based on its own cell state, $\mathbf{c}_i(t_k)$, and the states of its $\nu = 1, \dots, N$ direct neighbors, $\mathbf{c}_{i_\nu}(t_k)$, at a given time step t_k, which we formally collect in $\mathcal{N}_i = (\mathbf{c}_i(t_k), \mathbf{c}_{i_1}(t_k), \dots, \mathbf{c}_{i_N}(t_k))$. Instead, we utilize a trainable *sensory* ANN, $f_\theta^{(s)}(\cdot)$, that is applied individually to its own and every neighboring cell state, $\varepsilon_i(t_k) = f_\theta^{(s)}(\mathbf{c}_i(t_k))$ and $\varepsilon_{i_\nu}(t_k) = f_\theta^{(s)}(\mathbf{c}_{i_\nu}(t_k))$, to evaluate a sensor embedding, $\mathcal{E}(\mathcal{N}_i(t_k)) = (\varepsilon_i(t_k), \varepsilon_{i_1}(t_k), \dots, \varepsilon_{i_N}(t_k))$. The latter is averaged along the neighbor dimension to form a context vector $\mathbf{s}_i(t_k) = \langle \mathcal{E}(\mathcal{N}_i(t_k)) \rangle_\mathcal{N} \in \mathbb{R}^s$ of fixed size s that is permutation invariant with respect to the cell's neighborhood on the NCA (also see section II A). This context vector $\mathbf{s}_i(t_k)$ represents the cell's internal representation of its local environment on the cellular grid of the NCA.

Each cell i independently proposes an update, $\mathbf{a}_i(t_k)$, to its own state, $\mathbf{c}_i(t_k)$, potentially at every time step t_k following eq. (1). This update is computed by a controller ANN, $\mathbf{a}_i(t_k) = f_\theta^{(c)}(\mathbf{s}_i(t_k))$, based on the cell-specific context vector, $\mathbf{s}_i(t_k)$. Thus, the set of ANN parameters of the NCA comprises the sensory and controller network parameters, $f_\theta^{(s)}(\cdot)$ and $f_\theta^{(c)}(\cdot)$.

So far, we have not specified a particular architecture for either $f_\theta^{(s)}$ or $f_\theta^{(c)}$. Although the presented approach is agnostic to the particularly chosen ANN architecture, we here rely on rather simple implementations of ANNs: For the sensory ANN, $f_\theta^{(s)}$, we utilize a *Feed Forward* architecture with hyperbolic tangent activation function $\sigma(\cdot) = \tanh(\cdot)$, with 4 input units, 8 neurons in a single hidden layer, and 8 output neurons (defining the $(s = 8)$-dimensional context vector), resulting in a total of 112 parameters. For the controller ANN we utilize two different architectures, a *Feed Forward* (c.f., FF-agent in section III A and eq. (A1)) and a recurrent ANN that is inspired by both, *Recurrent* ANNs (RNNs)[108] and *Gene Regulatory Networks*[109] (c.f., RGRN-agent in section III A and eqs. (A2) and (A3) below).

The *Feed Forward* controller architecture consists of 8 input units (*i.e.*, the context vector from the sensory ANN), a single hidden layer with 6 neurons and a hyperbolic tangent activation function, and 4 output neurons without activation function, resulting in 82 parameters in total; thus the genuine *FF-agent* architecture in the main text comprises a total

number of $N_{\mathrm{FF}} = 194$ parameters (c.f., section III A). The *RGRN* controller architecture (see details below) consists of 8 input units, a single self-regulated recurrent state with 3 neurons (with an internal hyperbolic tangent activation), and 4 output neurons (without an activation function), resulting in 52 parameters in total; thus the genuine *RGRN-agent* architecture in the main text comprises a total number of $N_{\mathrm{RGRN}} = 164$ parameters (c.f., section III A). In table I we summarize the FF-agent and RGRN-agent's architectures and parameter counts.

	Sensory ANN (Num. Param.)	Controller ANN (Num. Param.)	Total Num. Param.
FF-agent	Feed Forward (112)	Feed Forward (82)	$112 + R \times 82$
RGRN-agent	Feed Forward (112)	RGRN (52)	$112 + R \times 52$

TABLE I: Architecture and number of parameters in sensory and controller ANNs of the two different agent architectures used in this contribution. The total number of parameters depends on the redundancy number R of the controller ANN (c.f., section III A).

Finally, we define the *RGRN* architecture, $\mathbf{y}(t_k) = f_\theta^{(\mathrm{RGRN})}(\mathbf{x}(t_k), \mathbf{h}(t_{k-1}))$, that relies on both an instantaneous input, $\mathbf{x}(t_k) \in \mathbb{R}^{\mathrm{I}}$, and a recurrent state, $\mathbf{h}(t_{k-1}) \in \mathbb{R}^{\mathrm{R}}$, from the previous iteration of the network to generate an output, $\mathbf{y}(t_k) \in \mathbb{R}^{\mathrm{O}}$: First, we define the self-regulated recurrent state $\mathbf{h}(t_k)$ as

$$\mathbf{h}(t_k) = (1 - \tau_1) \times \mathbf{h}(t_{k-1}) + \tau_2 \times \left[(\mathcal{U} \cdot \mathbf{x}(t_k) + \mathbf{b}_{\mathrm{U}}) + \tanh(\mathcal{V} \cdot \mathbf{h}(t_{k-1}) + \mathbf{b}_{\mathrm{V}})\right], \tag{A2}$$

which thus is maintained over time by a factor of $(1 - \tau_1)$ and updated by a factor of τ_2 via integrating external stimuli, $\mathbf{x}(t_k)$, and recurrent memory, $\mathbf{h}(t_{k-1})$, through the trainable matrices $\mathcal{U} \in \mathbb{R}^{\mathsf{H}\times\mathsf{I}}$, $\mathcal{V} \in \mathbb{R}^{\mathsf{H}\times\mathsf{H}}$ and bias vectors $\mathbf{b}_{\mathrm{U}}, \mathbf{b}_{\mathrm{V}} \in \mathbb{R}^H$, respectively. Second, we evaluate the network's output, $\mathbf{y}(t_k) \in \mathbb{R}^{\mathrm{O}}$, based on the RGRN's recurrent state $h(t_k)$, following

$$\mathbf{y}(t_k) = \sigma(\mathcal{W} \cdot \mathbf{h}(t_k) + \mathbf{b}_{\mathrm{W}}), \tag{A3}$$

having introduced the weight matrix $\mathcal{W} \in \mathbb{R}^{\mathsf{O}\times\mathsf{R}}$ and bias vector $\mathbf{b}_{\mathrm{W}} \in \mathbb{R}^{\mathsf{O}}$, and a non-linear activation function $\sigma(\cdot)$.

Following ideas from Ref. 109, we thus utilize with eq. (A2) an ANN that maintains a self-regulated (or "gene regulated") state, $\mathbf{h}(t_k)$. However - and dropping the bias vectors for convenience below - the second term in eq. (A2), *i.e.*, $[\mathcal{U} \cdot \mathbf{x}(t_k) + \tanh(\mathcal{V} \cdot \mathbf{h}(t_{k-1}))]$, is reminiscent to the kernel of an RNN[108], thus allowing the RGRN to integrate new information (*i.e.*, external stimuli) into its regulatory behavior. Thus, the state update of $\mathbf{h}(t_k)$ corresponds to regulating the network's recurrent state (or "gene expression") conditional to external stimuli. Furthermore, explicitly separating the self-regulated recurrent state from the RGRN's output allows us to utilize the RGRN as a controller, *i.e.*, to use its output, $\mathbf{y}(t_k)$, for updating the cell state of an NCA in eq. (1).

Here, we set $\tau_1 = 0.75$, $\tau_2 = 0.25$ and chose $\sigma(\cdot)$ as the identity transformation (*i.e.*, no, or linear activation of the RGRN's output), and we apply eq. (A2) 3 times (updating $\mathbf{h}(t_k)$ in every cycle) before forwarding the final value of $\mathbf{h}(t_k)$ to eq. (A3) to generate the RGRN's output.

For all ANN implementations we here relied on PyTorch[161].

Appendix B: A Reinforcement Learning Agent's Perception-Action Cycle

We utilize a Neural Cellular Automaton (NCA)[100] for morphogenesis tasks of two-dimensional target patterns. In such a setting, each cell of the NCA represents an autonomous agent that perceives details about its local environment (*i.e.*, the cell states of its direct neighbors on the NCA's spatial grid) and proposes actions to update its own state . Here, we summarize the terminology behind this perception-action cycle of an agent in an arbitrary environment of a Reinforcement Learning (RL) setting[113] (see fig. 1 (B) for an illustration):

Based on an agent's perception of the environment, i.e., a state $\mathbf{s}_k$ measured at time step t_k, the agent's goal is to manipulate the environment by taking an action $\mathbf{a}_k$ - resulting in a state update $\mathbf{s}_{k+1}$ in the next time step - to collect as much reward, $r_k \in \mathbb{R}$ (provided by the environment), as possible.

Formally, an agent picks its actions according to a policy $\pi_\theta(\mathbf{s}_{k' \leq k}) \rightarrow \mathbf{a}_k$, *i.e.*, a typically complicated function which might be parameterized via hidden variables θ. Artificial Neural Networks (ANNs) as universal function approximators[112] are promising candidates that can be *trained* to *fit* an agent's optimal policy (mapping states $\mathbf{s}_k$ to optimal actions $\mathbf{a}_k$[113]). Here, we thus utilize ANNs as a trainable update function of an NCA and deploy evolutionary algorithms (see section II B) to find the optimal policy (here, of morphogenesis tasks), $\pi_{\theta^*}(\mathbf{s}_{k' \leq k})$ via optimizing $\theta^* = \text{argmax}_\theta(\sum_{k' \leq k} r_{k'})$. This enables an agent to choose actions aiming at maximizing the expected cumulative reward (or maximum fitness, in our terms).

The particular functional choice of the reward signal defines the agent's task via positive (or negative) reinforcement. In our case, the cumulative reward, R_a of all N_a agents (*i.e.*, of all cells on the grid) after t_D time steps is summed up to the fitness $f = \sum R_a$ of the entire NCA. There is no general procedure for creating effective reward signals.

Crucially, we here do not provide the cellular agents with environmental reward feedback directly, but only use the cumulative reward, R_a, as a fitness criterion for the evolutionary algorithm. Thus, the collective of cells needs to evolve a signaling strategy to communicate desirable or prohibitive cell state updates during the corresponding developmental stage.

Appendix C: Fixed Boundary Condition Handling of the Neural Cellular Automaton

We employ Neural Cellular Automata (NCAs) with fixed boundary conditions on a two-dimensional square grid (see section II A). Each cell i is associated with integer grid-coordinates (x_i, y_i) on the NCA's grid of size $N_x \times N_y$, with $x_i \in [1, N_x]$ and $y_i \in [1, N_y]$.

The neighborhood of cell i is defined by all directly adjacent cells $i_{\nu=1,\ldots,N}$, *i.e.*, that share a border or a corner with cell i. Since we consider a square grid in this contribution, the grid coordinates of all $N = 8$ neighbor cells (x_{i_ν}, y_{i_ν}) are given by the permutations of $(x_i \pm m, y_i \pm n)$, with $m, n \in \{0, 1\}$, excluding the identity $m = n = 0$.

For cells at the boundaries of the grid, some neighbors with coordinates $x_{i_\nu}, y_{i_\nu} < 1$ or $> N_x, N_y$, respectively, will be out of bounds. Thus, we clip all neighbor coordinates to the intervals $[1, N_x]$ and $[1, N_y]$, respectively, via $x_{i_\nu} \rightarrow x_{i'_\nu} = \min(\max(x_{i_\nu}, 1), N_x)$, and $y_{i_\nu} \rightarrow y_{i'_\nu} = \min(\max(y_{i_\nu}, 1), N_y)$, and replace the neighbor index i_ν with the correspondingly index i'_ν of the respectively clipped coordinates $(x_{i'_\nu}, y_{i'_\nu})$. Collecting the numerical state values of the neighborhood of cell i thus yields $\mathcal{N}_i(t_k) \rightarrow \mathcal{N}'_i(t_k) = (\mathbf{c}_i(t_k), \mathbf{c}_{i'_1}(t_k), \ldots, \mathbf{c}_{i'_N}(t_k))$.

The matrix $\mathcal{N}_i'(t_k)$ then represents the input of the sensory part of the NCA's artificial neural network (c.f., section II A).

Appendix D: Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES)

Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES)[103] is a popular evolutionary algorithm: a multivariant normal distribution is utilized to model the (genotypic-) distribution of a set or a population of parameters that are evaluated against an objective function. Roughly speaking, this evaluated fitness score of an individual is associated with its probability of survival, and thus for participating in the reproduction of the next generation. The parameters of the multivariant normal distribution, *i.e.*, the mean and covariance matrix, are successively updated based on selecting the best individuals from a given population (or, more precisely, by weighting the relative importance of an individual by its fitness score) such that high-fitness individuals are generated with high likelihood by the Gaussian model. Thus, iteratively sampling "offspring" generations and adapting the model covariance matrix (and its mean) based on the population's fitness scores guides the evolutionary population toward high fitness regions in the parameter space over successive generations. Typically, also the numerical step size of the parameter update is adapted according to some inter- and intra-generation fitness measures. In a nutshell[103]:

- 1) CMA-ES typically starts with a standard (or parameterized) multi-variant normal distribution with the dimension given by the number of parameters (or genes).
- 2) At each evolutionary cycle, a new population of a fixed number of individuals is sampled from the model.
- 3) Each individual is evaluated against a fitness function, which quantifies the corresponding individual's probability of being selected for reproduction to form the next generation.
- 4) The mean and covariance matrix of the normal distribution, and a step-size parameter, are updated such that high-quality individuals are generated with high likelihood by the generative model.
- 5) The process (2-5) is repeated until a convergence criterion is met.

In the CMA-ES experiments presented in this contribution, we used an initial normal distribution with zero mean, $\mu_0 = 0$, and a standard deviation of typically $\sigma_0 = 2^{-4}$, and we disable step-size adaptation.

We specifically utilized the open-source *pycma* Python implementation of CMA-ES from Ref. 162.

Appendix E: Direct vs. Multi-scale Encoding: Morphogenetic Development over Evolutionary Time-Scales

In fig. 7, we explicitly illustrate the developmental process of the 8×8-Czech flag task over evolutionary time-scales for an NCA evolved at noise-level of $\xi_{\mathbf{c}} = 0.25$, decision-making probability $P_{\mathrm{D}} = 50\%$, and redundancy number $R = 4$; in fig. 8, we illustrate the same developmental process for an NCA without competency, *i.e.*, with $P_{\mathrm{D}} = 0$.

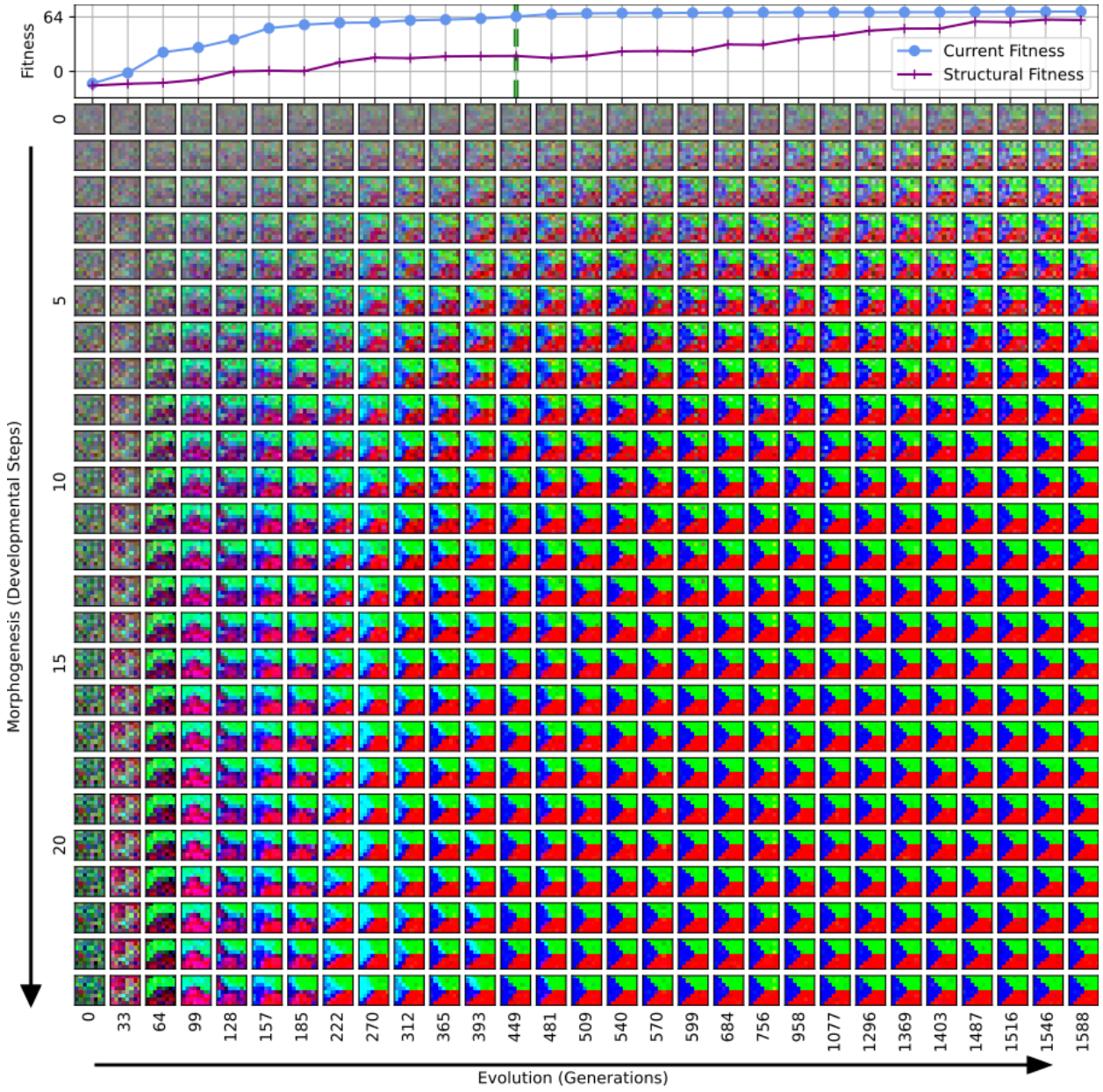

FIG. 7: The developmental process of the 8×8-Czech flag task (vertical axis) of selected generations over evolutionary time-scales (horizontal axis) for an NCA evolved with system parameters $\xi_{\mathbf{c}} = 0.25$, $P_{\mathrm{D}} = 50\%$, and $R = 4$. Each pixel corresponds to a cell of the NCA, at a given developmental step and generation, in an RGB notation corresponding to the numerical values of the first three cell states, scaled to values between $[0, 1]$. The top panel shows the current fitness of the respective generations (blue), and the structural fitness at $t_k = 0$ (purple); the green vertical dashed line marks the generation crossing the fitness threshold of $F_j = 64$ where we consider the problem solved.

These two figures illustrate how the evolutionary process learns how to construct the target pattern over generations, depending on the competency of the underlying substrate: either driven by intercellular communication-based self-assembly of the target pattern that continuously corrects potential errors of the developmental program, or via directly encoding the target pattern into the structural part of the genome to resist developmental noise for as long as possible. Moreover, while the target pattern for the competent first case is quickly (and robustly) self-assembled and maintained over time -

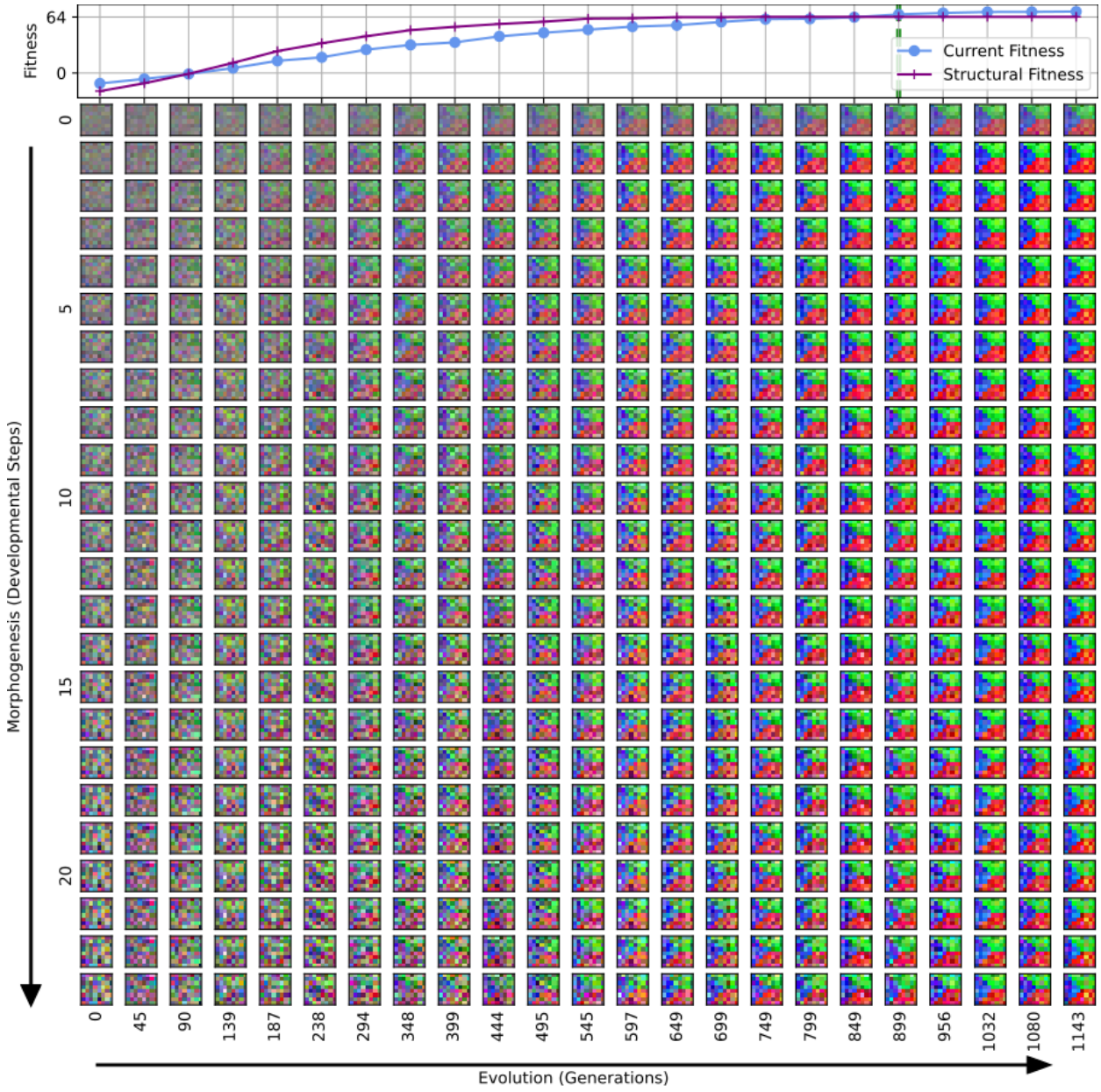

FIG. 8: Same as fig. 7 but for an NCA without competency (*i.e.*, $P_{\mathrm{D}} = 0$).

potentially much longer as the $t_{\mathrm{D}} = 25$ developmental time steps the phenotypes have been selected for - in the direct case the initial cell state eventually gets destroyed by the noise during the developmental process.

Thus, in the former case, illustrated in fig. 7, the structural fitness of the initial cell states (at $t_k = 0$) remains decoupled and rather low compared to the highest fitness of the population of the phenotypes, even long after the problem is solved. In contrast, in the latter case, illustrated in fig. 8, the initial cell state needs to evolve towards the target pattern directly, resulting in high structural fitness values at $t_k = 0$, which are then progressively decreased by the noise during the developmental process, resulting in correspondingly lower phenotypic fitness values.

Appendix F: Direct vs. Multi-scale Encoding: Evolution and Morphogenesis of a Smiley Face Pattern

In the main text, we primarily investigate the evolutionary implications of multi-scale intelligence on the example of morphogenesis of an 8×8 Czech flag pattern. To test, whether our findings in section III C generalize to different target patterns, we here present results for a much more involved task, namely a 9×9-smiley face pattern (c.f., fig. 1), which has several internal boundaries of (i) the face, (ii) the eyes, and (iii) the mouth; all other parameters are the same as for the 8×8 Czech flag task.

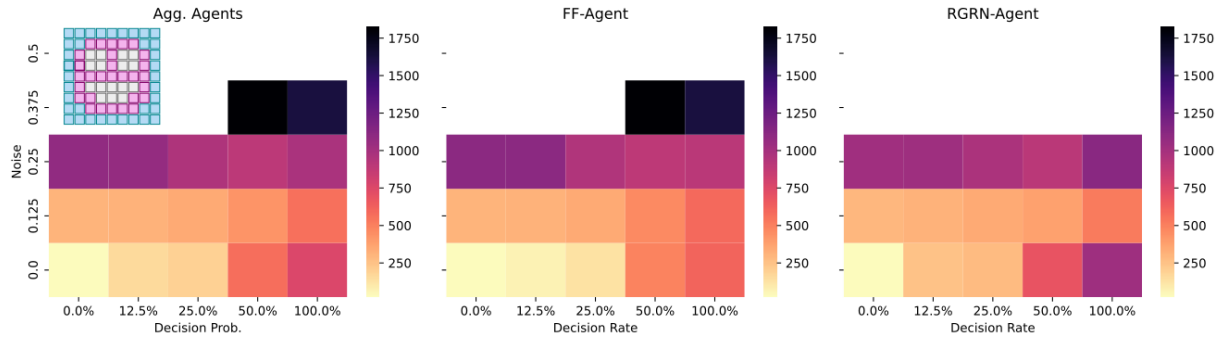

FIG. 9: Same as fig. 3 (C) but for a different target pattern, namely a 9×9 smiley face (inset in left panel). Moreover, we here aggregate over $R \geq 4$.

We thus perform an analogous study to section III C, and present the results in fig. 9 (reminiscent to fig. 3 (C)), but for redundancy numbers $R \geq 4$ (as we found that smaller controller networks perform systematically worse on the task, suggesting a capacity bottleneck of ANNs with $R < 4$ in this case). Analogously to the much simpler 8×8 Czech task, we can learn from fig. 9 that, while in the low noise regime, direct encoding can lead to a more efficient evolutionary process, in situations with increasing developmental noise higher competency levels (here again realized via the decision-making probability) can significantly enhance the efficiency of the evolutionary process of a morphogenesis task. Notably, and due to computational reasons, we evaluated only two to three independent evolutionary processes for every combination of the system parameters (noise-level, decision-making probability, and redundancy number) for the results depicted in fig. 9. However, the overall trend of the evolutionary efficiency of (i) directly encoding the target pattern and (ii) encoding the functional parameters of a multi-scale competency architecture is consistent with our previous results discussed in section III C. Due to the increased complexity of the 9×9-smiley face task, the critical noise level that separates the evolutionary efficiency of (i) and (ii) is correspondingly shifted to larger values of here $\xi_{\mathbf{c}} \geq 0.25$ (c.f., fig. 3 (C)).

Analogous to fig. 7, we illustrate in fig. 10 the developmental process of the 9×9-smiley face task over evolutionary time-scales for an NCA evolved at noise-level of $\xi_{\mathbf{c}} = 0.25$, decision-making probability $P_{\mathrm{D}} = 50\%$, and redundancy number of $R = 4$ in an RGB scheme attributing the numerical values of the first three cell states, scaled to values between $[0, 1]$, respectively (c.f., section II A).

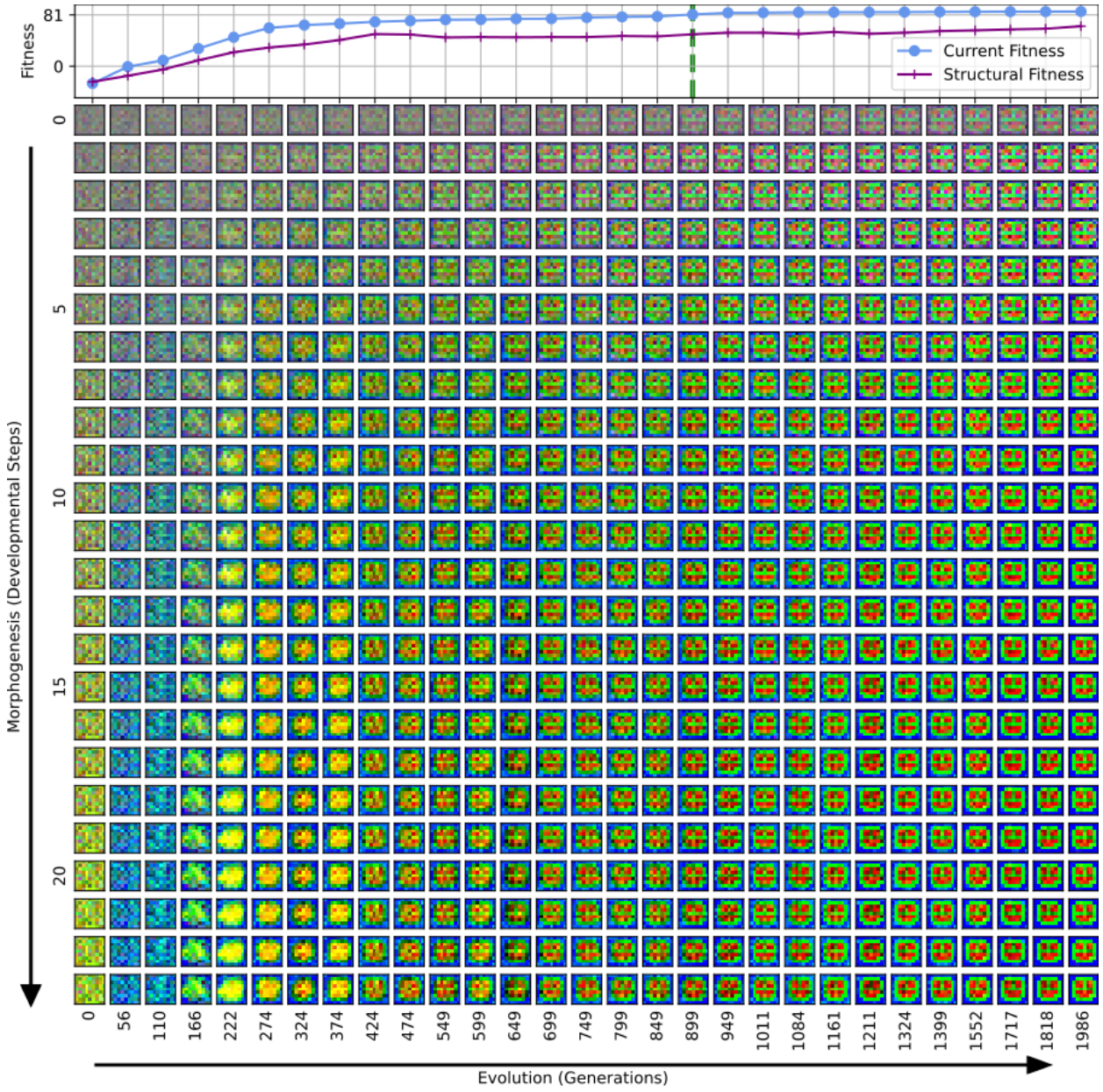

FIG. 10: Same as fig. 7 but for the 9×9-Smiley face task, with a fitness threshold of $F_j = 81$ (c.f., green vertical dashed line).

Appendix G: Evolution exploits Redundancy at the Cost of a More Complex Search Space

Analogous to section III C, we here present the evolutionary efficiency of the same morphogenesis experiments of the 8×8 Czech flag problem depicted in fig. 3, but present as a function of the redundancy number R - instead of the decision-making probability P_{D} - and the noise level $\xi_{\mathbf{c}}$; for a given combination of R and $\xi_{\mathbf{c}}$, we additionally utilized different values for $P_{\mathrm{D}} = \{0.25, 0.5, 1.0\}$ and performed 15 statistically independent runs of the EA for each parameter combination, resulting in 45 independent evolutionary runs per $(R, \xi_{\mathbf{c}})$-tuple.

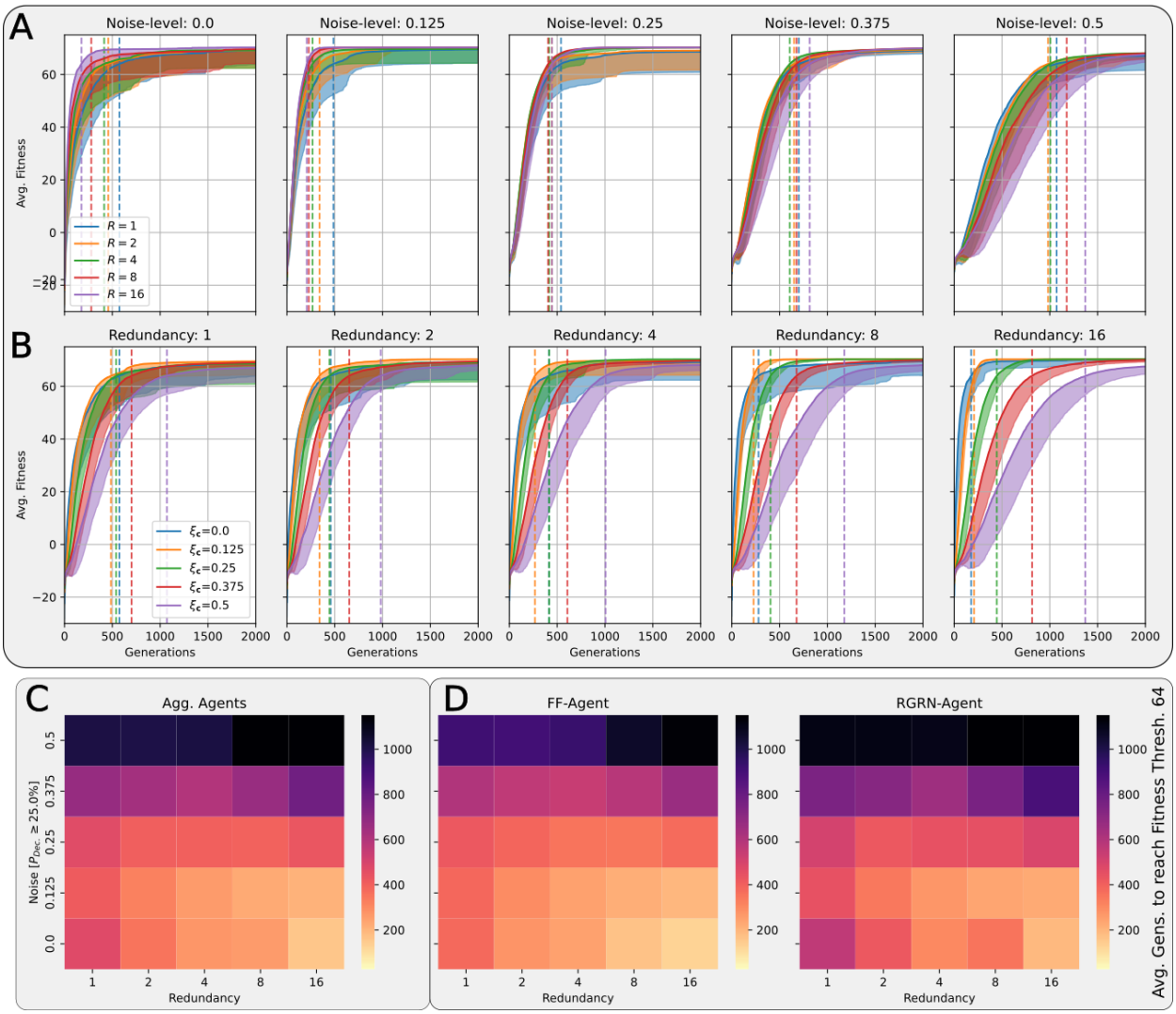

FIG. 11: Same as fig. 3 but presented for the redundancy number R vs. noise level $\xi_{\mathbf{c}}$ and aggregated over all values of the decision-making probability $P_{\mathrm{D}} \geq 0.25$.

Although there appears to be an effect of R on the evolutionary efficiency (c.f., panels A and C, D of fig. 11), the results are less pronounced compared to fig. 11. Despite considerable uncertainty in the evolutionary efficiency, as shown in fig. 11 (A, B), we can learn from the heatmaps, fig. 11 (C, D), that at low noise levels of $\xi_{\mathbf{c}} = 0$ or 0.125, large R values appear favorable over lower ones, whereas, at larger noise levels of $\xi_{\mathbf{c}} = 0.5$, populations with lower values of R perform better on average. For intermediate noise levels of $\xi_{\mathbf{c}} = 0.25$ and 0.375, we observe an "optimal" redundancy number of 4, in this particular example.

This suggests a trade-off in the evolutionary efficiency of redundancy - as an affordance of competency - and the corresponding increase in the number of overall parameters of the functional genome.

Appendix H: Morphogenesis at Scale with a hybrid Compositional Pattern-Producing Network - Neural Cellular Automata Model

The particular choice of especially the structural part of the genome $\mathbf{x}_j^{(\mathrm{S})}$ in eq. (3) limits the scalability of our multi-scale competency approach of morphogenesis to significantly larger systems, as the size of the structural genome will grow correspondingly with the number of cells in the system. However, by utilizing Compositional Pattern Producing Networks (CPPNs)[120,138] the parameters, θ_H, of a hyper-network, $f_\theta^{(\mathrm{H})}(\cdot)$, could replace the structural genes in eq. (3) such that the initial states of each cell i are indirectly encoded by the hyper-network based on their relative spatial positions, $(x_i/N_x, y_i/N_y)$, on the Neural Cellular Automaton's (NCA's) grid via $\mathbf{c}_i(0) = f_\theta^{(\mathrm{H})}(x_i/N_x, y_i/N_y)$.

However, it has proven to be difficult, if not numerically infeasible, to reliably and exactly reproduce a two-dimensional target pattern using CPPNs[139]. Thus, we here propose a hybrid approach for morphogenesis at scale of a CPPN indirectly encoding the initial cell states of an NCA, whose uni-cellular agents are then challenged to self-assemble the desired target pattern in a morphogenetic developmental stage. This would allow for scaling the target pattern arbitrarily either during training or during deployment since the number of cells on the NCA's grid does not affect the size of the (structural part of) genome.

REFERENCES

[1] M. Levin, "Darwin's agential materials: evolutionary implications of multiscale competency in developmental biology," Cellular and Molecular Life Sciences **80**, 142 (2023).

[2] C. Fields and M. Levin, "Regulative development as a model for origin of life and artificial life studies," Biosystems **229**, 104927 (2023).

[3] C. Fields and M. Levin, "Competency in navigating arbitrary spaces as an invariant for analyzing cognition in diverse embodiments," Entropy **24**, 819 (2022).

[4] M. Levin, "Collective Intelligence of Morphogenesis as a Teleonomic Process," in *Evolution "On Purpose": Teleonomy in Living Systems* (The MIT Press, 2023).

[5] K. D. Birnbaum and A. S. Alvarado, "Slicing across kingdoms: Regeneration in plants and animals," Cell **132**, 697–710 (2008).

[6] A. K. Harris, "The need for a concept of shape homeostasis," Biosystems **173**, 65–72 (2018).

[7] M. Levin, A. M. Pietak, and J. Bischof, "Planarian regeneration as a model of anatomical homeostasis: Recent progress in biophysical and computational approaches," Seminars in Cell & Developmental Biology **87**, 125–144 (2019).

[8] L. N. Vandenberg, D. S. Adams, and M. Levin, "Normalized shape and location of perturbed craniofacial structures in the xenopus tadpole reveal an innate ability to achieve correct morphology," Developmental Dynamics **241**, 863–878 (2012).

[9] J. Cooke, "Scale of body pattern adjusts to available cell number in amphibian embryos," Nature **290**, 775–778 (1981).

[10] G. Fankhauser, "Maintenance of normal structure in heteroploid salamander larvae, through compensation of changes in cell size by adjustment of cell number and cell shape," Journal of Experimental Zoology **100**, 445–455 (1945).

[11] D. J. Blackiston and M. Levin, "Ectopic eyes outside the head in Xenopus tadpoles

provide sensory data for light-mediated learning," Journal of Experimental Biology **216**, 1031–1040 (2013).
[12]G. Pezzulo and M. Levin, "Re-membering the body: applications of computational neuroscience to the top-down control of regeneration of limbs and other complex organs," Integrative Biology **7**, 1487–1517 (2015).
[13]J. Davies and M. Levin, "Synthetic morphology with agential materials," Nature Reviews Bioengineering **1**, 46–59 (2023).
[14]J. M. Baldwin, "A new factor in evolution," The American Naturalist **30**, 441–451 (1896).
[15]W. H. Thorpe, "Animal learning and evolution," Nature **156**, 46–46 (1945).
[16]G. G. Simpson, "The baldwin effect," Evolution **7**, 110 (1953).
[17]G. E. Hinton, S. J. Nowlan, *et al.*, "How learning can guide evolution," Complex Systems **1**, 495–502 (1987).
[18]R. K. Belew, "When both individuals and populations search: Adding simple learning to the genetic algorithm," in *Proceedings of the 3rd International Conference on Genetic Algorithms* (Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1989) p. 34–41.
[19]D. Ackley and M. Littman, "Interactions between learning and evolution," in *Proceedings of the Second Conference on Artificial Life*, Vol. 10, edited by C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen (Addison-Wesley, Santa Fe, NM, USA, 1991) pp. 487–509.
[20]G. Mayley, "Landscapes, Learning Costs, and Genetic Assimilation," Evolutionary Computation **4**, 213–234 (1996).
[21]L. Bull, "On the Baldwin Effect," Artificial Life **5**, 241–246 (1999).
[22]S. Nolfi and D. Floreano, "Learning and evolution," Autonomous Robots **7**, 89–113 (1999).
[23]H. Dopazo, "A model for the interaction of learning and evolution," Bulletin of Mathematical Biology **63**, 117–134 (2001).
[24]B. H. Weber and D. J. Depew, *Evolution and Learning: The Baldwin Effect Reconsidered* (The MIT Press, 2003).
[25]F. Mery and T. J. Kawecki, "A fitness cost of learning ability in drosophila melanogaster," Proceedings of the Royal Society of London. Series B: Biological Sciences **270**, 2465–2469 (2003).
[26]E. Crispo, "The Baldwin effect and genetic assimilation: revisiting two mechanisms of evolutionary change mediated by phenotypic plasticity," Evolution **61**, 2469–2479 (2007).
[27]T. J. Kawecki, "Evolutionary ecology of learning: insights from fruit flies," Population Ecology **52**, 15–25 (2009).
[28]I. Paenke, T. J. Kawecki, and B. Sendhoff, "The influence of learning on evolution: A mathematical framework," Artificial Life **15**, 227–245 (2009).
[29]K. M. Hoedjes, H. M. Kruidhof, M. E. Huigens, M. Dicke, L. E. M. Vet, and H. M. Smid, "Natural variation in learning rate and memory dynamics in parasitoid wasps: opportunities for converging ecology and neuroscience," Proceedings of the Royal Society B: Biological Sciences **278**, 889–897 (2010).
[30]R. A. Watson and E. Szathmáry, "How can evolution learn?" Trends in Ecology & Evolution **31**, 147–157 (2016).
[31]A. Livnat and C. Papadimitriou, "Evolution and learning: Used together, fused together. a response to watson and szathmáry," Trends in Ecology & Evolution **31**, 894–896

(2016).
[32]C. Fields, J. Bischof, and M. Levin, "Morphological coordination: A common ancestral function unifying neural and non-neural signaling," Physiology **35**, 16–30 (2020), pMID: 31799909.
[33]F. Baluška and S. Mancuso, "Deep evolutionary origins of neurobiology: Turning the essence of 'neural' upside-down," Communicative & Integrative Biology **2**, 60–65 (2009), pMID: 19513267.
[34]F. Baluška, W. B. Miller, and A. S. Reber, "Cellular and evolutionary perspectives on organismal cognition: from unicellular to multicellular organisms," Biological Journal of the Linnean Society **139**, 503–513 (2022).
[35]L. Shreesha and M. Levin, "Cellular competency during development alters evolutionary dynamics in an artificial embryogeny model," Entropy **25** (2023), 10.3390/e25010131.
[36]V. Liard, D. P. Parsons, J. Rouzaud-Cornabas, and G. Beslon, "The Complexity Ratchet: Stronger than Selection, Stronger than Evolvability, Weaker than Robustness," Artificial Life **26**, 38–57 (2020).
[37]J. Huizinga, K. O. Stanley, and J. Clune, "The Emergence of Canalization and Evolvability in an Open-Ended, Interactive Evolutionary System," Artificial Life **24**, 157–181 (2018).
[38]W. C. Ratcliff, J. D. Fankhauser, D. W. Rogers, D. Greig, and M. Travisano, "Origins of multicellular evolvability in snowflake yeast," Nature Communications **6** (2015), 10.1038/ncomms7102.
[39]J. L. Payne, J. H. Moore, and A. Wagner, "Robustness, Evolvability, and the Logic of Genetic Regulation," Artificial Life **20**, 111–126 (2014).
[40]K. Raman and A. Wagner, "The evolvability of programmable hardware," Journal of The Royal Society Interface **8**, 269–281 (2011).
[41]C. L. Nehaniv, "Evolvability," Biosystems **69**, 77–81 (2003).
[42]H. Hoenigsberg, "Cell biology, molecular embryology, lamarckian and darwinian selection as evolvability," Genet Mol Res **2**, 7–28 (2003).
[43]T. F. Hansen, "Is modularity necessary for evolvability?" Biosystems **69**, 83–94 (2003).
[44]M. A. Bedau and N. H. Packard, "Evolution of evolvability via adaptation of mutation rates," Biosystems **69**, 143–162 (2003).
[45]M. Kirschner and J. Gerhart, "Evolvability," Proceedings of the National Academy of Sciences **95**, 8420–8427 (1998).
[46]G. P. Wagner and L. Altenberg, "Perspective: Complex adaptations and the evolution of evolvability," Evolution **50**, 967–976 (1996).
[47]R. Raff, "Developmental mechanisms in the evolution of animal form: origins and evolvability of body plans," Early life on earth **84**, 489–500 (1994).
[48]R. Raff, *The Shape of Life: Genes, Development, and the Evolution of Animal Form*, The Shape of Life: Genes, Development, and the Evolution of Animal Form (University of Chicago Press, 1996).
[49]P. Alberch, "From genes to phenotype: dynamical systems and evolvability," Genetica **84**, 5–11 (1991).
[50]S. A. Frank, "Natural selection maximizes Fisher information," Journal of Evolutionary Biology **22**, 231–244 (2009).
[51]S. A. Frank, "Maladaptation and the paradox of robustness in evolution," PLoS ONE **2**, e1021 (2007).

[52]S. A. Frank, "Developmental selection and self-organization," Biosystems **40**, 237–243 (1997).
[53]S. A. Frank, "The design of adaptive systems: Optimal parameters for variation and selection in learning and development," Journal of Theoretical Biology **184**, 31–39 (1997).
[54]K. O. Stanley and R. Miikkulainen, "A Taxonomy for Artificial Embryogeny," Artificial Life **9**, 93–130 (2003).
[55]C. Waddington, "The strategy of the genes," L.: Allen and Unwin (1957).
[56]C. H. Waddington, *The strategy of the genes* (Routledge, 2014).
[57]D. Noble, "Modern physiology vindicates darwin's dream," Experimental Physiology **107**, 1015–1028 (2022).
[58]J. A. Shapiro, "Engines of innovation: biological origins of genome evolution," Biological Journal of the Linnean Society **139**, 441–456 (2022).
[59]A. Szilágyi, P. Szabó, M. Santos, and E. Szathmáry, "Phenotypes to remember: Evolutionary developmental memory capacity and robustness," PLOS Computational Biology **16**, e1008425 (2020).
[60]E. Szathmáry, "Toward major evolutionary transitions theory 2.0," Proceedings of the National Academy of Sciences **112**, 10104–10111 (2015).
[61]C. L. Buckley and R. A. Watson, (2022, manuscript in preperation).
[62]K. Kouvaris, J. Clune, L. Kounios, M. Brede, and R. A. Watson, "How evolution learns to generalise: Using the principles of learning theory to understand the evolution of developmental organisation," PLOS Computational Biology **13**, e1005358 (2017).
[63]E. Fox Keller, "Elusive locus of control in biological development: Genetic versus developmental programs," Journal of Experimental Zoology **285**, 283–290 (1999).
[64]E. Jablonka, "The evolutionary implications of epigenetic inheritance," Interface Focus **7**, 20160135 (2017).
[65]K. Laland, T. Uller, M. Feldman, K. Sterelny, G. B. Müller, A. Moczek, E. Jablonka, J. Odling-Smee, G. A. Wray, H. E. Hoekstra, D. J. Futuyma, R. E. Lenski, T. F. C. Mackay, D. Schluter, and J. E. Strassmann, "Does evolutionary theory need a rethink?" Nature **514**, 161–164 (2014).
[66]M. Elgart, O. Snir, and Y. Soen, "Stress-mediated tuning of developmental robustness and plasticity in flies," Biochimica et Biophysica Acta (BBA) - Gene Regulatory Mechanisms **1849**, 462–466 (2015).
[67]O. T. Eldakar and D. S. Wilson, "Eight Criticisms not to make about Group Selection," Evolution **65**, 1523–1526 (2011).
[68]S. Karve and A. Wagner, "Environmental complexity is more important than mutation in driving the evolution of latent novel traits in e. coli," Nature Communications **13**, 5904 (2022).
[69]S. Karve and A. Wagner, "Multiple novel traits without immediate benefits originate in bacteria evolving on single antibiotics," Molecular Biology and Evolution **39**, msab341 (2022).
[70]A. Wagner, "The molecular origins of evolutionary innovations," Trends in Genetics **27**, 397–410 (2011).
[71]A. Wagner and W. Rosen, "Spaces of the possible: universal darwinism and the wall between technological and biological innovation," Journal of The Royal Society Interface **11**, 20131190 (2014).
[72]D. S. Wilson, "A theory of group selection." Proceedings of the National Academy of Sciences **72**, 143–146 (1975).

[73]R. Calabretta, A. D. Ferdinando, G. P. Wagner, and D. Parisi, "What does it take to evolve behaviorally complex organisms?" Biosystems **69**, 245–262 (2003).
[74]G. Schlosser and G. P. Wagner, eds., *Modularity in Development and Evolution* (University of Chicago Press, Chicago, IL, 2004).
[75]B. M. Stadler, P. F. Stadler, G. P. Wagner, and W. Fontana, "The topology of the possible: Formal spaces underlying patterns of evolutionary change," Journal of Theoretical Biology **213**, 241–274 (2001).
[76]K. H. Ten Tusscher and P. Hogeweg, "Evolution of networks for body plan patterning; interplay of modularity, robustness and evolvability," PLoS Computational Biology **7**, e1002208 (2011).
[77]G. P. Wagner, M. Pavlicev, and J. M. Cheverud, "The road to modularity," Nature Reviews Genetics **8**, 921–931 (2007).
[78]G. P. WAGNER and P. F. STADLER, "Quasi-independence, homology and the unity of type: A topological theory of characters," Journal of Theoretical Biology **220**, 505–527 (2003).
[79]D. A. Power, R. A. Watson, E. Szathmáry, R. Mills, S. T. Powers, C. P. Doncaster, and B. Czapp, "What can ecosystems learn? expanding evolutionary ecology with learning theory," Biology Direct **10** (2015), 10.1186/s13062-015-0094-1.
[80]H. I. Schreier, Y. Soen, and N. Brenner, "Exploratory adaptation in large random networks," Nature Communications **8**, 14826 (2017).
[81]Y. Soen, M. Knafo, and M. Elgart, "A principle of organization which facilitates broad lamarckian-like adaptations by improvisation," Biology Direct **10**, 68 (2015).
[82]T. Uller, A. P. Moczek, R. A. Watson, P. M. Brakefield, and K. N. Laland, "Developmental Bias and Evolution: A Regulatory Network Perspective," Genetics **209**, 949–966 (2018).
[83]R. A. Watson, R. Mills, and C. L. Buckley, "Global Adaptation in Networks of Selfish Components: Emergent Associative Memory at the System Scale," Artificial Life **17**, 147–166 (2011).
[84]R. A. Watson, R. Mills, C. L. Buckley, K. Kouvaris, A. Jackson, S. T. Powers, C. Cox, S. Tudge, A. Davies, L. Kounios, and D. Power, "Evolutionary connectionism: Algorithmic principles underlying the evolution of biological organisation in evo-devo, evo-eco and evolutionary transitions," Evolutionary Biology **43**, 553–581 (2016).
[85]R. A. Watson, G. P. Wagner, M. Pavlicev, D. M. Weinreich, and R. Mills, "The evolution of phenotypic correlations and "developmental memory"," Evolution **68**, 1124–1138 (2014).
[86]J. Bongard, "Morphological change in machines accelerates the evolution of robust behavior," Proceedings of the National Academy of Sciences **108**, 1234–1239 (2011).
[87]J. Bongard, C. Laschi, H. Lipson, N. Cheney, and F. Corucci, "Material properties affect evolution's ability to exploit morphological computation in growing soft-bodied creatures," in *Artificial Life Conference Proceedings* (MIT press One Rogers Street, Cambridge, MA 02142-1209, USA, 2016) pp. 234–241.
[88]S. Kriegman, N. Cheney, and J. Bongard, "How morphological development can guide evolution," Scientific Reports **8**, 13934 (2018).
[89]M. Levin, "Bioelectric networks: the cognitive glue enabling evolutionary scaling from physiology to mind," Animal Cognition **26**, 1865–1891 (2023).
[90]G. Pezzulo and M. Levin, "Top-down models in biology: explanation and control of complex living systems above the molecular level," Journal of The Royal Society Interface

13, 20160555 (2016).
[91]V. Nanos and M. Levin, "Multi-scale chimerism: An experimental window on the algorithms of anatomical control," Cells & Development **169**, 203764 (2022).
[92]D. Lobo, M. Solano, G. A. Bubenik, and M. Levin, "A linear-encoding model explains the variability of the target morphology in regeneration," Journal of The Royal Society Interface **11**, 20130918 (2014).
[93]M. Levin, "The Computational Boundary of a "Self": Developmental Bioelectricity Drives Multicellularity and Scale-Free Cognition," Frontiers in Psychology **10** (2019), 10.3389/fpsyg.2019.02688.
[94]M. Levin, "Technological approach to mind everywhere: An experimentally-grounded framework for understanding diverse bodies and minds," Frontiers in Systems Neuroscience **16** (2022), 10.3389/fnsys.2022.768201.
[95]L. Pio-Lopez, J. Bischof, J. V. LaPalme, and M. Levin, "The scaling of goals from cellular to anatomical homeostasis: an evolutionary simulation, experiment and analysis," Interface Focus **13**, 20220072 (2023).
[96]C. G. Langton, *Artificial life: An overview* (Mit Press, 1997).
[97]J. Von Neumann, A. W. Burks, *et al.*, "Theory of self-reproducing automata," IEEE Transactions on Neural Networks **5**, 3–14 (1966).
[98]M. Games, "The fantastic combinations of john conway's new solitaire game "life" by martin gardner," Sci. Am. **223**, 120–123 (1970).
[99]B. W.-C. Chan, "Lenia: Biology of artificial life," Complex Systems **28**, 251–286 (2019).
[100]X. Li and A. G.-O. Yeh, "Neural-network-based cellular automata for simulating multiple land use changes using gis," Int. J. Geogr. Inf. Sci. **16**, 323–343 (2002).
[101]A. Mordvintsev, E. Randazzo, E. Niklasson, and M. Levin, "Growing neural cellular automata," Distill **5** (2020), 10.23915/distill.00023.
[102]S. Pontes-Filho, K. Walker, E. Najarro, S. Nichele, and S. Risi, "A unified substrate for body-brain co-evolution," in *From Cells to Societies: Collective Learning across Scales* (2022).
[103]E. Hansen and G. W. Walster, *Global Optimization using Interval Analysis: Revised and Expanded*, 2nd ed. (CRC Press, 2004).
[104]L. N. Vandenberg, R. D. Morrie, and D. S. Adams, "V-atpase-dependent ectodermal voltage and ph regionalization are required for craniofacial morphogenesis," Developmental Dynamics **240**, 1889–1904 (2011).
[105]R. W. Hamming, "Error detecting and error correcting codes," The Bell System Technical Journal **29**, 147–160 (1950).
[106]C. Wang, D. Sklar, and D. Johnson, "Forward error-correction coding," Crosslink **3**, 26–29 (2001).
[107]Y. Zhang, M. C. Fontaine, V. Bhatt, S. Nikolaidis, and J. Li, "Arbitrarily scalable environment generators via neural cellular automata," arXiv preprint arXiv:2310.18622 (2023).
[108]D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations* (MIT Press, Cambridge, MA, USA, 1986) p. 318–362.
[109]T. W. Hiscock, "Adapting machine-learning algorithms to design gene circuits," BMC Bioinformatics **20**, 214 (2019).
[110]S. Wolfram, *A New Kind of Science* (Wolfram Media, 2002).

[111]M. Cook, "Universality in elementary cellular automata," Complex Syst. **15** (2004).
[112]K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," Neural networks **2**, 359–366 (1989).
[113]R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction* (The MIT Press, 2018).
[114]Y. Tang and D. Ha, "The sensory neuron as a transformer: Permutation-invariant neural networks for reinforcement learning," in *Advances in Neural Information Processing Systems*, edited by A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan (2021).
[115]L. Wolpert, "Chapter 6 positional information and pattern formation," in *Current Topics in Developmental Biology Volume 6* (Elsevier, 1971) p. 183–224.
[116]J. Sharpe, "Wolpert's French Flag: what's the problem?" Development **146**, dev185967 (2019).
[117]M. Gabalda-Sagarra, L. B. Carey, and J. Garcia-Ojalvo, "Recurrence-based information processing in gene regulatory networks," Chaos: An Interdisciplinary Journal of Nonlinear Science **28**, 106313 (2018).
[118]S. Biswas, S. Manicka, E. Hoel, and M. Levin, "Gene regulatory networks exhibit several kinds of memory: Quantification of memory in biological and random transcriptional networks," iScience **24**, 102131 (2021).
[119]S. Biswas, W. Clawson, and M. Levin, "Learning in transcriptional network models: Computational discovery of pathway-level memory and effective interventions," International Journal of Molecular Sciences **24**, 285 (2022).
[120]M. Etcheverry, C. Moulin-Frier, P.-Y. Oudeyer, and M. Levin, "Ai-driven automated discovery tools reveal diverse behavioral competencies of biological networks," (2023), 10.31219/osf.io/s6thq.
[121]The terminology FF and RGRN stems from the respective agents' *Feed Forward* and *Recurrent Gene Regulatory Network* ANN controller layers (see appendix A for details).
[122]K. O. Stanley and R. Miikkulainen, "A taxonomy for artificial embryogeny," Artificial life **9**, 93–130 (2003).
[123]J. Bongard and M. Levin, "There's plenty of room right here: Biological systems as evolved, overloaded, multi-scale machines," Biomimetics **8**, 110 (2023).
[124]The L_2 regularization applied to $\mathbf{x}_j$ does not introduce a bias between minimal, $P_{\mathrm{D},j} = 0$, and maximal competency levels, $P_{\mathrm{D},j} = 1$, as the L_2 regularization is applied to $\mathbf{x}_j^{(\mathrm{C})}$, not $P_{\mathrm{D},j}$; both $P_{\mathrm{D},j}$ and the L_2 regularization are symmetric with respect to the sign of $\mathbf{x}_j^{(\mathrm{C})}$.
[125]C. Gershenson and D. Helbing, "When slower is faster," Complexity **21**, 9–15 (2015).
[126]We model and investigate the evolution of the process of morphogenesis and morphostasis *in silico* and deploy our framework to a self-orchestrated pattern formation task, primarily of a two-dimensional, 8×8 Czech flag pattern but also for other, much more-involved target shapes such as a 9×9 smiley face (see appendix F).
[127]A. Mordvintsev, E. Randazzo, and C. Fouts, "Growing isotropic neural cellular automata," in *The 2022 Conference on Artificial Life*, ALIFE 2022 (MIT Press, 2022).
[128]J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *Proceedings of the 32nd International Conference on Machine Learning*, Proceedings of Machine Learning Research, Vol. 37, edited by F. Bach and D. Blei (PMLR, Lille, France, 2015) pp. 2256–2265.

[129]J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," Advances in neural information processing systems **33**, 6840–6851 (2020).
[130]A. Q. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," in *Proceedings of the 38th International Conference on Machine Learning*, Proceedings of Machine Learning Research, Vol. 139, edited by M. Meila and T. Zhang (PMLR, 2021) pp. 8162–8171.
[131]P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," in *Advances in Neural Information Processing Systems*, Vol. 34, edited by M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan (Curran Associates, Inc., 2021) pp. 8780–8794.
[132]J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," (2022), arXiv:2010.02502 [cs.LG].
[133]D. Savic, B. Belintzev, L. Beloussov, and A. Zaraisky, "Morphogenetic activity prepattern in embryonic epithelia," Progress in clinical and biological research **217A**, 101—104 (1986).
[134]P. Hunt and R. Krumlauf, "Deciphering the hox code: Clues to patterning branchial regions of the head," Cell **66**, 1075–1078 (1991).
[135]H. L. Ashe and J. Briscoe, "The interpretation of morphogen gradients," Development **133**, 385–394 (2006).
[136]M. Levin and C. J. Martyniuk, "The bioelectric code: An ancient computational medium for dynamic control of growth and form," Biosystems **164**, 76–93 (2018).
[137]R. B. Palm, M. G. Duque, S. Sudhakaran, and S. Risi, "Variational neural cellular automata," in *International Conference on Learning Representations* (2022).
[138]K. O. Stanley, "Compositional pattern producing networks: A novel abstraction of development," Genetic Programming and Evolvable Machines **8**, 131–162 (2007).
[139]B. G. Woolley and K. O. Stanley, "On the deleterious effects of a priori objectives on evolution and representation," in *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, GECCO '11 (ACM, 2011).
[140]B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: representing scenes as neural radiance fields for view synthesis," Commun. ACM **65**, 99–106 (2021).
[141]T. Parr, G. Pezzulo, and K. J. Friston, *Active Inference: The Free Energy Principle in Mind, Brain, and Behavior* (The MIT Press, 2022).
[142]D. Ha and J. Schmidhuber, "Recurrent world models facilitate policy evolution," in *Advances in Neural Information Processing Systems*, Vol. 31, edited by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Curran Associates, Inc., 2018).
[143]S. Pontes-Filho, P. G. Lind, and S. Nichele, "Assessing the robustness of critical behavior in stochastic cellular automata," Physica D: Nonlinear Phenomena **441**, 133507 (2022).
[144]J. M. Beggs, "The criticality hypothesis: how local cortical networks might optimize information processing," Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **366**, 329–343 (2008).
[145]W. L. Shew and D. Plenz, "The functional benefits of criticality in the cortex," The Neuroscientist **19**, 88–100 (2012).
[146]W. L. Shew, W. P. Clawson, J. Pobst, Y. Karimipanah, N. C. Wright, and R. Wessel, "Adaptation to sensory input tunes visual cortex to criticality," Nature Physics **11**, 659–

663 (2015).
[147]W. P. Clawson, N. C. Wright, R. Wessel, and W. L. Shew, "Adaptation towards scale-free dynamics improves cortical stimulus discrimination at the cost of reduced detection," PLOS Computational Biology **13**, e1005574 (2017).
[148]F. Habibollahi, B. J. Kagan, A. N. Burkitt, and C. French, "Critical dynamics arise during structured information presentation within embodied in vitro neuronal networks," Nature Communications **14** (2023), 10.1038/s41467-023-41020-3.
[149]S. A. Jones, J. H. Barfield, V. K. Norman, and W. L. Shew, "Scale-free behavioral dynamics directly linked with scale-free cortical dynamics," eLife **12**, e79950 (2023).
[150]A. D. Wissner-Gross and C. E. Freer, "Causal entropic forces," Phys. Rev. Lett. **110**, 168702 (2013).
[151]L. Pio-Lopez and M. Levin, "Aging as a morphostasis defect: a developmental bioelectricity perspective," (2023), 10.31219/osf.io/wkhx4.
[152]W. B. Miller Jr, F. Baluška, A. S. Reber, and P. Slijepčević, "Why death and aging ? all memories are imperfect," Progress in Biophysics and Molecular Biology (2024), https://doi.org/10.1016/j.pbiomolbio.2024.02.001.
[153]W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," The Bulletin of Mathematical Biophysics **5**, 115–133 (1943).
[154]M. L. Minsky, *Theory of neural-analog reinforcement systems and its application to the brain-model problem* (Princeton University, 1954).
[155]M. Minsky, "Steps toward artificial intelligence," Proceedings of the IRE **49**, 8–30 (1961).
[156]F. Rosenblatt *et al.*, *Principles of neurodynamics: Perceptrons and the theory of brain mechanisms*, Vol. 55 (Spartan books Washington, DC, 1962).
[157]M. Minsky and S. Papert, "Perceptron: an introduction to computational geometry," The MIT Press, Cambridge, expanded edition **19**, 2 (1969).
[158]D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," nature **323**, 533–536 (1986).
[159]Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature **521**, 436–444 (2015).
[160]I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, 2016).
[161]A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32* (Curran Associates, Inc., 2019) pp. 8024–8035.
[162]N. Hansen, yoshihikoueno, ARF1, G. Kadlecová, K. Nozawa, L. Rolshoven, M. Chan, Y. Akimoto, brieglhostis, and D. Brockhoff, "Cma-es/pycma: r3.3.0," (2023).