Environmental variability and network structure determine the optimal plasticity mechanisms in embodied agents

Emmanouil Giannakakis[1,2], Sina Khajehabdollahi [1] and Anna Levina[1,2,3]

[1]Department of Computer Science, University of Tübingen, Tübingen, Germany
[2]Max Planck Institute for Biological Cybernetics, Tübingen, Germany
[3]Bernstein Center for Computational Neuroscience Tübingen, Tübingen, Germany

Abstract

The evolutionary balance between innate and learned behaviors is highly intricate, and different organisms have found different solutions to this problem. We hypothesize that the emergence and exact form of learning behaviors is naturally connected with the statistics of environmental fluctuations and tasks an organism needs to solve. Here, we study how different aspects of simulated environments shape an evolved synaptic plasticity rule in static and moving artificial agents. We demonstrate that environmental fluctuation and uncertainty control the reliance of artificial organisms on plasticity. Interestingly, the form of the emerging plasticity rule is additionally determined by the details of the task the artificial organisms are aiming to solve. Moreover, we show that co-evolution between static connectivity and interacting plasticity mechanisms in distinct sub-networks changes the function and form of the emerging plasticity rules in embodied agents performing a foraging task.

Introduction

One of the defining features of living organisms is their ability to adapt to their environment and incorporate new information to modify their behavior. It is unclear how the ability to learn first evolved (Papini, 2012), but its utility appears evident. Natural environments are too complex for all the necessary information to be hardcoded genetically (Snell-Rood, 2013) and more importantly, they keep changing during an organism's lifetime in ways that cannot be anticipated (Ellefsen, 2014; Dunlap and Stephens, 2016). The link between learning and environmental uncertainty and fluctuation has been extensively demonstrated in both natural (Kerr and Feldman, 2003; Snell-Rood and Steck, 2019), and artificial environments (Nolfi and Parisi, 1996).

Nevertheless, the ability to learn does not come without costs. For the capacity to learn to be beneficial in evolutionary terms, a costly nurturing period is often required, a phenomenon observed in both biological (Thornton and Clutton-Brock, 2011), and artificial organisms (Eskridge and Hougen, 2012). Additionally, it has been shown that in some complex environments, hardcoded behaviors may be superior to learned ones given limits in the agent's lifetime and environmental uncertainty (Dunlap and Stephens, 2009; Fawcett et al., 2012; Lange and Sprekeler, 2020).

The theoretical investigation of the optimal balance between learned and innate behaviors in natural and artificial systems goes back several decades. However, it has recently found also a wide range of applications in applied AI systems (Lee and Lee, 2020; Biesialska et al., 2020). Most AI systems are trained for specific tasks, and have no need for modification after their training has been completed. Still, technological advances and the necessity to solve broad families of tasks make discussions about life-like AI systems relevant to a wide range of potential application areas. Thus the idea of open-ended AI agents (Open Ended Learning Team et al., 2021) that can continually interact with and adapt to changing environments has become particularly appealing.

Many different approaches for introducing lifelong learning in artificial agents have been proposed. Some of them draw direct inspiration from actual biological systems (Schmidhuber, 1987; Parisi et al., 2019). Among them, the most biologically plausible solution is to equip artificial neural networks with some local neural plasticity (Thangarasa et al., 2020), similar to the large variety of synaptic plasticity mechanisms (Citri and Malenka, 2008; Feldman, 2009; Caroni et al., 2012) that performs the bulk of the learning in the brains of living organisms (Magee and Grienberger, 2020). The artificial plasticity mechanisms can be optimized to modify the connectivity of the artificial neural networks toward solving a particular task. The optimization can use a variety of approaches, most commonly evolutionary computation.

The idea of meta-learning or optimizing synaptic plasticity rules to perform specific functions has been recently established as an engineering tool that can compete with state-of-the-art machine learning algorithms on various complex tasks (Burms et al., 2015; Najarro and Risi, 2020; Pedersen and Risi, 2021; Yaman et al., 2021). Additionally, it can be used to reverse engineer actual plasticity mechanisms found in biological neural networks and uncover their functions (Confavreux et al., 2020; Jordan et al., 2021).

Here, we study the effect that different factors (environmental fluctuation and reliability, task complexity) have on the form of evolved functional reward-modulated plasticity rules. We investigate the evolution of plasticity rules in static, single-layer simple networks. Then we increase the complexity by switching to moving agents performing a complex foraging task. In both cases, we study the impact of different environmental parameters on the form of the evolved plasticity mechanisms and the interaction of learned and static network connectivity. Interestingly, we find that different environmental conditions and different combinations of static and plastic connectivity have a very large impact on the resulting plasticity rules.

Methods

Environment

We imagine an agent who must forage to survive in an environment presenting various types of complex food particles. Each food particle is composed of various amounts and combinations of N ingredients that can have positive (food) or negative (poison) values. The value of a food particle is a weighted sum of its ingredients. To predict the reward value of a given resource, the agent must learn the values of these ingredients by interacting with the environment. The priors could be generated by genetic memory, but the exact values are subject to change.

To introduce environmental variability, we stochastically change the values of the ingredients. More precisely, we define two ingredient-value distributions E_1 and E_2 (Guttenberg, 2019) and switch between them, with probability p_{tr} for every time step. We control how (dis)similar the environments are by parametrically setting $E_2 = (1-2d_e)E_1$, with $d_e \in [0,1]$ serving as a distance proxy for the environments; when $d_e = 0$, the environment remains unchanged, and when $d_e = 1$ the value of each ingredient fully reverses when the environmental transition happens. For simplicity, we take values of the ingredients in E_1 equally spaced between -1 and 1 (for the visualization, see Fig. 3a, b).

Static agent

The static agent receives passively presented food as a vector of ingredients and can assess its compound value using the linear summation of its sensors with the (learned or evolved) weights, see Fig. 1. The network consists of N sensory neurons that are projecting to a single post-synaptic neuron. At each time step, an input $X_t = (x_1, \ldots, x_N)$ is presented, were the value x_i, $i \in \{1, \ldots, N\}$ represents the quantity of the ingredient i. We draw x_i independently form a uniform distribution on the $[0,1]$ interval ($x_i \sim U(0,1)$). The value of each ingredient w_i^c is determined by the environment (E_1 or E_2).

The postsynaptic neuron outputs a prediction of the food X_t value as $y_t = g(WX_t^T)$. Throughout the paper, g will be either the identity function, in which case the prediction neuron is linear, or a step-function; however, it could be any other nonlinearity, such as a sigmoid or ReLU. After outputting the prediction, the neuron receives feedback in the form of the real value of the input R_t. The real value is computed as $R_t = W^c X_t^T + \xi$, where $W^c = (w_1^c, \ldots, w_N^c)$ is the actual value of the ingredients, and ξ is a term summarizing the noise of reward and sensing system $\xi \sim \mathcal{N}(0, \sigma)$.

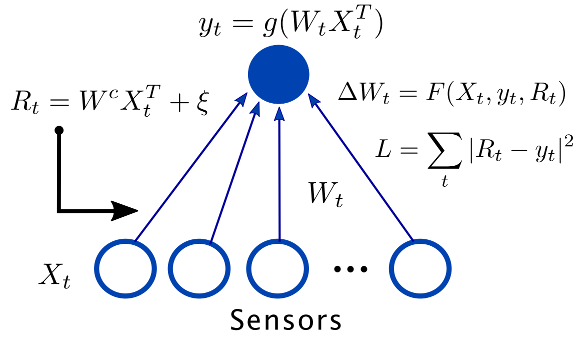

Figure 1: *An outline of the static agent's network. The sensor layer receives inputs representing the quantity of each ingredient of a given food at each time step. The agent computes the prediction of the food's value y_t and is then given the true value R_t; it finally uses this information in the plasticity rule to update the weight matrix.*

For the evolutionary adjustment of the agent's parameters, the loss of the static agent is the sum of the mean squared errors (MSE) between its prediction y_t and the reward R_t over the lifetime of the agent. The agent's initial weights are set to the average of the two ingredient value distributions, which is the optimal initial value for the case of symmetric switching of environments that we consider here.

Moving Agent

As a next step, we incorporate the sensory network of static agents into embodied agents that can move around in an environment scattered with food. To this end, we merge the static agent's network with a second, non-plastic motor network that is responsible for controlling the motion of the agent in the environment. Specifically, the original plastic network now provides the agent with information about the value of the nearest food. The embodied agent has additional sensors for the distance from the nearest food, the angle between the current velocity and the nearest food direction, its own velocity, and its own energy level (sum of consumed food values). These inputs are processed by two hidden layers (of 30 and 15 neurons) with $\tanh$ activation. The network's outputs are angular and linear acceleration, Fig. 2.

The embodied agents spawn in a 2D space with periodic boundary conditions along with a number of food particles that are selected such that the mean of the food value distribution is ~ 0. An agent can eat food by approaching it sufficiently closely, and each time a food particle is eaten,

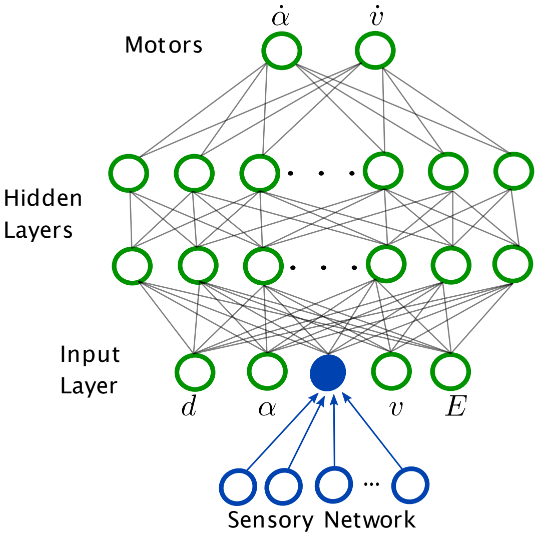

Figure 2: *An outline of the network controlling the foraging agent. The sensor layer receives inputs at each time step (the ingredients of the nearest food), which are processed by the plastic layer in the same way as the static sensory network, Fig. 1. The output of that network is given as input to the motor network, along with the distance d and angle α to the nearest food, the current velocity v, and energy E of the agent. These signals are processed through two hidden layers to the final output of motor commands as the linear and angular acceleration of the agent*

it is re-spawned with the same value somewhere randomly on the grid (following the setup of (Khajehabdollahi et al., 2022)). After 5000 time steps, the cumulative reward of the agent (the sum of the values of all the food it consumed) is taken as its fitness, at which point the best agents are selected by the genetic algorithm and used to initialize the next generation. The environment (food and agents' positions) is re-initialized at the start of each generation. During the evolutionary optimization, the parameters for both the motor network (connections) and plastic network (learning rule parameters) are evolved simultaneously (the genotype includes both motor weights and plasticity parameters), and so agents must learn to move and discriminate good/bad food at the same time.

Plasticity rule parametrization

Reward-modulated plasticity is one of the most promising explanations for biological credit assignment (Legenstein et al., 2008). In our network, the plasticity rule that updates the weights of the linear sensor network is a reward-modulated rule which is parameterized as a linear combination of the input, the output, and the reward at each time step:

$$\Delta W_t = \eta_p[R_t \cdot \overbrace{(\theta_1 X_t y_t + \theta_2 y_t + \theta_3 X_t + \theta_4)}^{\text{Reward Modulated}} + \underbrace{(\theta_5 X_t y_t + \theta_6 y_t + \theta_7 X_t + \theta_8)}_{\text{Hebbian}}]. \qquad (1)$$

Additionally, after each plasticity step, the weights are normalized by mean subtraction, an important step for the stabilization of Hebbian-like plasticity rules (Zenke and Gerstner, 2017).

We use a genetic algorithm to optimize the learning rate η_p and amplitudes of different terms $\theta = (\theta_1, \ldots, \theta_8)$. The successful plasticity rule after many food presentations must converge to a weight vector that predicts the correct food values (or allows the agent to correctly decide whether to eat a food or avoid it).

To have comparable results, we divide $\theta = (\theta_1, \ldots, \theta_8)$ by $\theta_{\max} = \max_k |\theta_k|$. So that $\theta/\theta_{\max} = \theta^{\text{norm}} \in [-1, 1]^8$. We then multiply the learning rate η_p with $\theta_{\max}$ to maintain the rule's evolved form unchanged, $\eta_p^{\text{norm}} = \eta_p \cdot \theta_{\max}$. In the following, we always use normalized η_p and θ, omitting $^{\text{norm}}$.

Evolutionary Algorithm

To evolve the plasticity rule and the moving agents' motor networks, we use a simple genetic algorithm with elitism (Deb, 2011). The agents' parameters are initialized at random (drawn from a Gaussian distribution), then the sensory network is trained by the plasticity rule and finally, the agents are evaluated. After each generation, the best-performing agents (top 10 % of the population size) are selected and copied into the next generation. The remaining 90 % of the generation is repopulated with mutated copies of the best-performing agents. We mutate agents by adding independent Gaussian noise ($\sigma = 0.1$) to its parameters. Unless specified otherwise, we train a population of 100 agents for 200 generations.

Results

Environmental and reward variability control the evolved learning rates of the static agents

To start with, we consider a static agent whose goal is to identify the value of presented food correctly. The static reward-prediction network quickly evolves the parameters of the learning rule, successfully solving the prediction task. We first look at the evolved learning rate η_p, which determines how fast (if at all) the network's weight vector is updated during the lifetime of the agents. We identify three factors that control the learning rate parameter the EA converges to: the distance between the environments, the noisiness of the reward, and the rate of environmental transition.

The first natural factor is the distance d_e between the two environments, with a larger distance requiring a higher

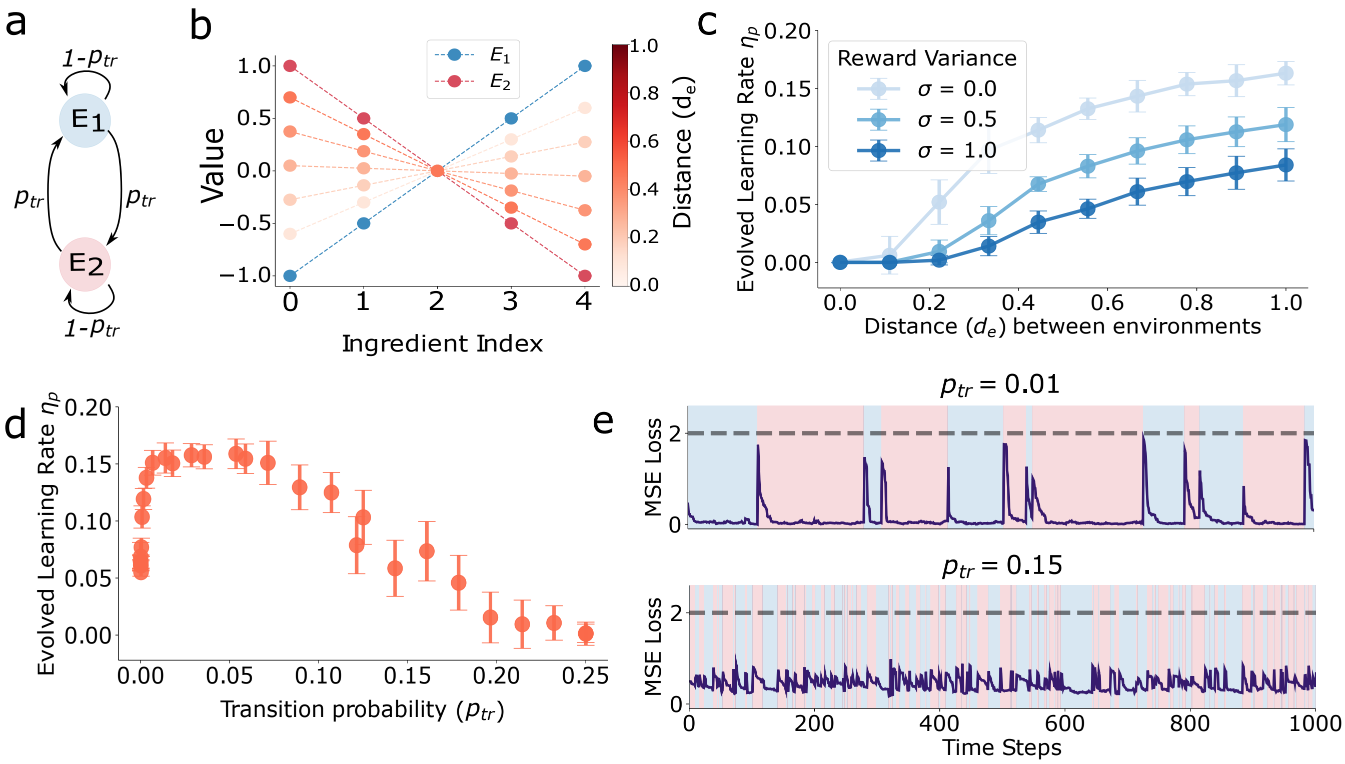

Figure 3: ***a.*** *Schematic representation of two-states Markov model with transition probability* p_{tr} *between two environments* E_1 *and* E_2 *defined by the ingredient value distributions.* ***b.*** *We vary the* E_2 *environment by changing the ingredient values linearly* $E_2 = (1 - 2d_e)E_1$*, the* d_e *is indicated by the color.* ***c.*** *The evolved learning rate* η_p *grows with the distance* d_e *between the environments and decreases with the reward variance* σ*.* ***d.*** *The environment transition probability* p_{tr} *(here for* $d_e = 1$ *and* $\sigma = 0.25$*) has a non-monotonous relationship with the evolved learning rate* η_p*. Up to a certain point, more rapid transitions lead to faster learning, but too rapid environmental transition leads to a reduction of the evolved learning rate.* ***e.*** *For slow environmental transition (top), the agent fully adapts to the environment after each transition. If the transitions happen fast (bottom), the agent maintains an intermediate position between the two environments and never fully adapts to either of them.*

learning rate, Fig. 3c. This is an expected result since the convergence time to the "correct" weights is highly dependent on the initial conditions. If an agent is born at a point very close to optimality, which naturally happens if the environments are similar, the distance it needs to traverse on the fitness landscape is small. Therefore it can afford to have a small learning rate, which leads to a more stable convergence and is not affected by noise.

A second parameter that impacts the learning rate is the variance of the rewards. The reward an agent receives for the plasticity step contains a noise term ξ that is drawn from a zero mean Gaussian distribution with standard deviation σ. This parameter controls the unreliability of the agent's sensory system, i.e., higher σ means that the information the agent gets about the value of the foods it consumes cannot be fully trusted to reflect the actual value of the foods. As σ increases, the learning rate η_p decreases, which means that the more unreliable an environment becomes, the less an agent relies on plasticity to update its weights, Fig. 3c. Indeed for some combinations of relatively small distance d_e and high reward variance σ, the EA converges to a learning rate of $\eta_p \approx 0$. This means that the agent opts to have no adaptation during its lifetime and remain at the mean of the two environments. It is an optimal solution when the expected loss due to ignoring the environmental transitions is, on average, lower than the loss the plastic network will incur by learning via the (often misleading because of the high σ) environmental cues.

A final factor that affects the learning rate the EA will converge to is the frequency of environmental change during an agent's lifetime. Since the environmental change is modeled as a simple, two-state Markov process (Fig. 3a), the control parameter is the transition probability p_{tr}.

When keeping everything else the same, the learning rate rapidly rises as we increase the transition probability from 0, and after reaching a peak, it begins to decline slowly, eventually reaching zero (Fig. 3d). This means that when environmental transition is very rare, agents opt for a very

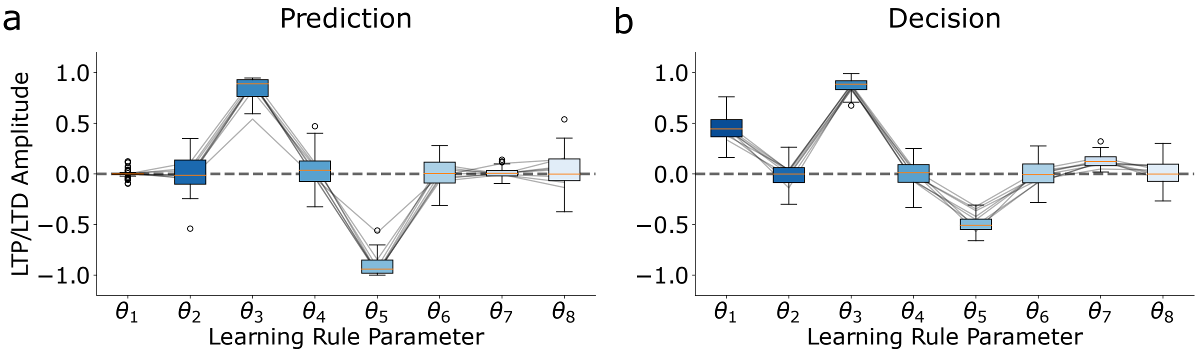

Figure 4: *The evolved parameters $\theta = (\theta_1, \ldots, \theta_8)$ of the plasticity rule for the reward prediction (**a.**) and the decision (**b.**) tasks, for a variety of parameters ($p_{tr} = 0.01$, $d_e \in 0, 0.1, \ldots, 1$, and $\sigma \in 0, 0.1, \ldots, 1$ in all 100 combinations). Despite the relatively small difference between the tasks, the evolved learning rules differ considerably. For visual guidance, the lines connect θs from the same run.*

low learning rate, allowing a slow and stable convergence to an environment-appropriate weight vector that leads to very low losses while the agent remains in that environment. As the rate of environmental transition increases, faster learning is required to speed up convergence in order to exploit the (comparatively shorter) stays in each environment. Finally, as the environmental transition becomes too fast, the agents opt for slower or even no learning, which keeps them near the middle of the two environments, ensuring that the average loss of the two environments is minimal (Fig. 3d).

The form of the evolved learning rule depends on the task: Decision vs. Prediction

The plasticity parameters $\theta = (\theta_1, \ldots, \theta_8)$ for the reward-prediction task converge on approximately the same point, regardless of the environmental parameters (Fig. 4a). In particular, $\theta_3 \to 1$, $\theta_5 \to -1$, $\theta_i \to 0$ for all other i, and thus the learning rule converges to:

$$\Delta W_t = \eta_p[\theta_3 X_t R_t + \theta_5 X_t y_t] \approx \eta_p X_t(R_t - y_t). \tag{2}$$

Since by definition $y_t = g(W_t X_t^T) = W_t X_t^T$ ($g(x) = x$ in this experiment) and $R_t = W^c X_t^T + \xi$ we get:

$$\Delta W_t = \eta_p X_t (W^c - W_t) X_t^T + \eta_p \xi X_t^T. \tag{3}$$

Thus the distribution of ΔW_t converges to a distribution with mean 0 and variance depending on η_p and σ and W converges to W^c. So this learning rule will match the agent's weight vector with the vector of ingredient values in the environment.

We examine the robustness of the learning rule the EA discovers by considering a slight modification of our task. Instead of predicting the expected food value, the agent now needs to decide whether to eat the presented food or not. This is done by introducing a step-function nonlinearity ($g(x) = 1$ if $x \geq 1$ and 0 otherwise). Then the output $y(t)$ is computed as:

$$y_t = \begin{cases} 1, & \text{if } W_t X_t^T \geq 0, \\ 0, & \text{if } W_t X_t^T < 0. \end{cases} \tag{4}$$

Instead of the MSE loss between prediction and actual value, the fitness of the agent is now defined as the sum of the food values it chose to consume (by giving $y_t = 1$). Besides these two changes, the setup of the experiments remains exactly the same.

The qualitative relation between η_p and parameters of environment d_e, σ and p_{tr} is preserved in the changed experiment. However, the resulting learning rule is significantly different (Fig. 4). The evolution converges to the following learning rule:

$$\Delta W_t = \begin{cases} \eta_p X_t[\theta_3 R_t + \theta_7], \ y_t = 0, \\ \eta_p X_t[(\theta_1 + \theta_3) R_t + (\theta_5 + \theta_7)], \ y_t = 1. \end{cases} \tag{5}$$

In both cases, the rule has the form $\Delta W_t = \eta_p X_t[\alpha_y R_t + \beta_y]$. Thus, the ΔW_t is positive or negative depending on whether the reward R_t is above or below a threshold ($\gamma = -\beta_y / \alpha_y$) that depends on the output decision of the network ($y_t = 0$ or 1).

Both learning rules have a clear Hebbian form (coordination of pre- and post-synaptic activity) and use the incoming reward signal as a threshold. These similarities indicate some common organizing principles of reward-modulated learning rules, but their significant differences highlight the sensitivity of the optimization process to task details.

The learning rate of embodied agents depends on environmental variability

We now turn to the moving embodied agents in the 2D environment. To optimize these agents, both the motor network's connections and the sensory network's plasticity parameters

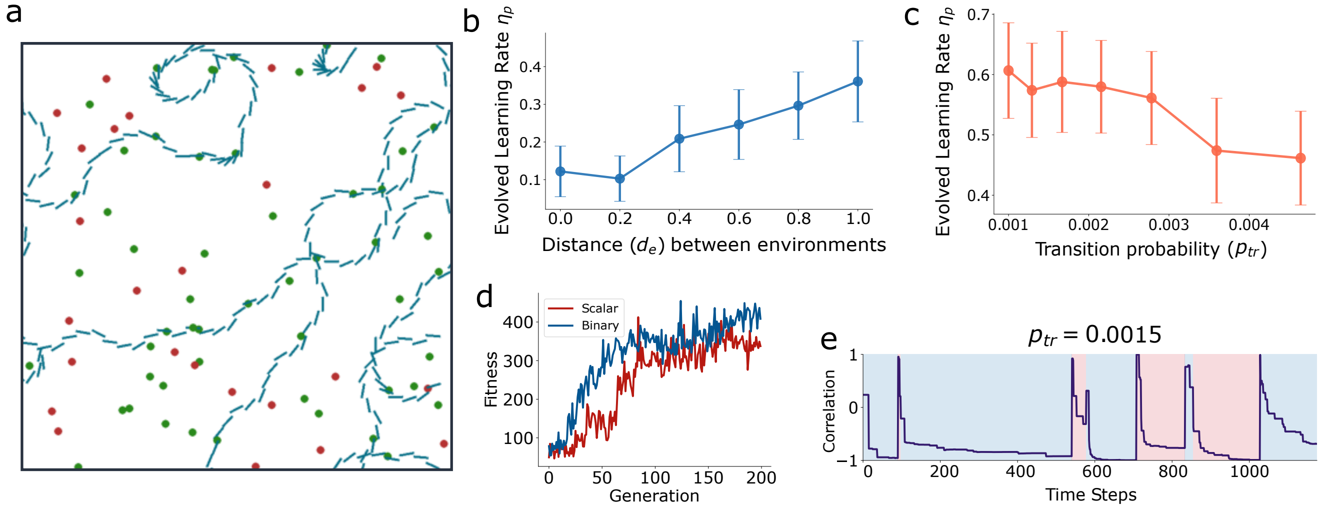

Figure 5: ***a.*** *The trajectory of an agent (blue line) in the 2D environment. A well-trained agent will approach and consume food with positive values (green dots) and avoid negative food (red dots).* ***b.*** *The learning rate of the plastic sensory network* eta_p *grows with the distance between environments* d_e ***c.*** *and decreases with the frequency of environmental change.* ***d.*** *The fitness of an agent (measured as the total food consumed over its lifetime) increases over generations of the EA for both the scalar and binary readouts in the sensory network.* ***e.*** *The Pearson correlation coefficient of an evolved agent's weights with the ingredient value vector of the current environment (*E_1 *- blue,* E_2 *- red). In this example, the agent's weights are anti-correlated with its environment, which is not an issue for performance since the motor network can interpret the inverted signs of food.*

evolve simultaneously. Since the motor network is initially random and the agent has to move to find food, the number of interactions an agent experiences in its lifetime can be small, slowing down the learning. However, having the larger motor network also has benefits for evolution because it allows the output of the plastic network to be read out and transformed in different ways, resulting in a broad set of solutions.

The agents can solve the task effectively by evolving a functional motor network and a plasticity rule that converges to interpretable weights (Fig. 5a). After $\sim$ 100 evolutionary steps (Fig. 5d), the agents can learn the ingredient value distribution using the plastic network and reliably move towards foods with positive values while avoiding the ones with negative values.

We compare the dependence of the moving and the static agents on the parameters of the environment: d_e and the state transition probability p_{tr}. At first, in order to simplify the experiment, we set the transition probability to 0, but fixed the initial weights to be the average of E_1 and E_2, while the real state is E_2. In this experiment, the distance between states d_e indicates twice the distance between the agent's initial weights and the optimal weights (the environment's ingredient values) since the agent is initialized at the mean of the two environment distributions. Same as for the static agent, the learning rate increases with the distance d_e (Fig. 5b).

Then, we examine the effect of the environmental transition probability p_{tr} on the evolved learning rate η_p. In order for an agent to get sufficient exposure to each environment, we scale down the probability p_{tr} from the equivalent experiment for the static agents. We find that as the probability of transition increases, the evolved learning rate η_p decreases (Fig. 5c). This fits with the larger trend for the static agent, although there is a clear difference when it comes to the increase for very small transition probabilities that were clearly identifiable in the static but not the moving agents. This could be due to much sparser data and possibly the insufficiently long lifetime of the moving agent (the necessity of scaling makes direct comparisons difficult). Nevertheless, overall we see that the associations observed in the static agents between environmental distance d_e and transition probability p_{tr} and the evolved learning rate η_p are largely maintained in the moving agents. Still, more data would be needed to make any conclusive assertions about the exact effect of these environmental parameters on the emerging plasticity mechanisms.

Rule redundancy in the embodied agents

A crucial difference between the static and the moving agents is the function the plasticity has to perform. While in the static agents, the plasticity has to effectively identify the exact value distribution of the environment in order to produce accurate predictions, in the embodied agents, the plasticity has to merely produce a representation of the environment that the motor network can evolve to interpret

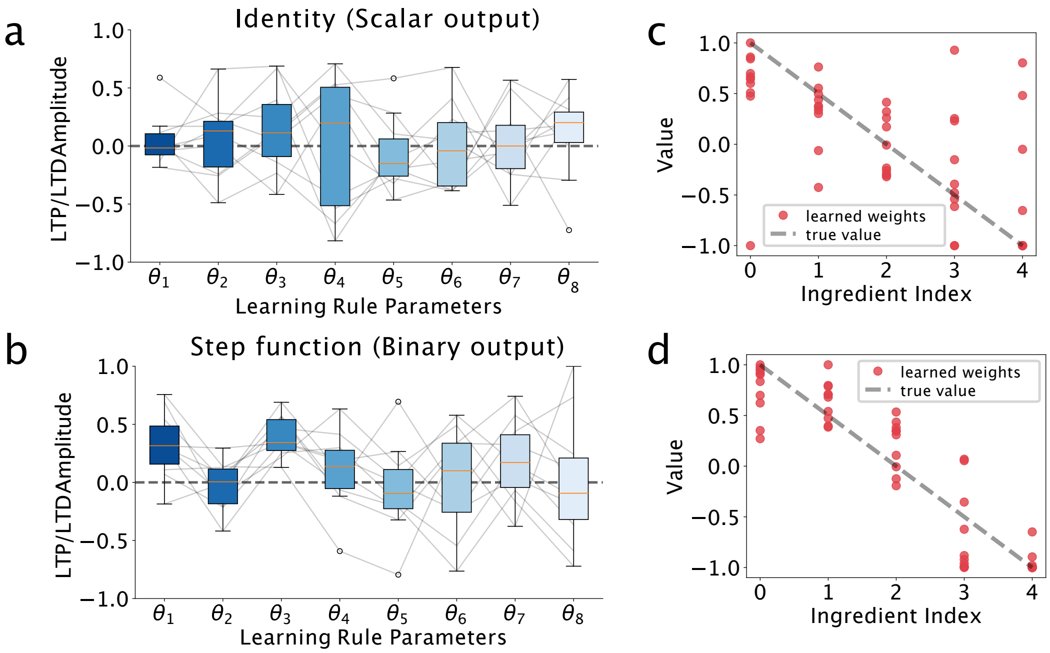

Figure 6: *The evolved parameters of moving agents' plasticity rule for the* $g(s) = x$, *identity (**a.**) and the step function (Eq. 4) (**b.**) sensory networks (the environmental parameters here are* $d_e \in [0, 1]$, $\sigma = 0$ *and* $p_{tr} = 0.001$*). The step function (binary output) network evolved a more structured plasticity rule (e.g.,* $\theta_3 > 0$ *for all realizations) than the linear network. Moreover, the learned weights for the identity network (**c.**) have higher variance and correlate significantly less with the environment's ingredient distribution compared to the learned weights for the thresholded network (**d.**)*

adequately enough to make decisions about which food to consume.

To illustrate the difference, we plot the Pearson correlation coefficient between an agent's weights and the ingredient values of the environment it is moving in (Fig. 5e). We use the correlation instead of the MSE loss (which we used for the static agents in Fig. 3e) because the amplitude of the weight vector varies a lot for different agents and meaningful conclusions cannot be drawn from the MSE loss. For many agents, the learned weights are consistently anti-correlated with the actual ingredient values (an example of such an agent is shown in Fig. 5e). This means that the output of the sensory network will have the opposite sign from the actual food value. While in the static network, this would lead to very bad predictions and high loss, in the foraging task, these agents perform exactly as well as the ones where the weights and ingredients values are positively correlated, since the motor network can simply learn to move towards food for which it gets a negative instead of a positive sensory input.

This additional step of the output of the plastic network going through the motor network before producing any behavior has a strong effect on the plasticity rules that the embodied agents evolve. Specifically, if we look at the emerging rules the top performing agents have evolved (Fig. 6a), it becomes clear that, unlike the very well-structured rules of the static agents (Fig. 4a), there is now virtually no discernible pattern or structure. The difference becomes even clearer if we look at the learned weights (at the end of a simulation) of the best-performing agents (Fig. 6c). While there is some correlation with the environment's ingredient value distribution, the variance is very large, and they do not seem to converge on the "correct" values in any way. This is to some extent expected since, unlike the static agents where the network's output has to be exactly correct, driving the evolution of rules that converge to the precise environmental distribution, in the embodied networks, the bulk of the processing is done by the motor network which can evolve to interpret the scalar value of the sensory network's output in a variety of ways. Thus, as long as the sensory network's plasticity rule co-evolves with the motor network, any plasticity rule that learns to produce consistent information about the value of encountered food can potentially be selected.

To further test this assumption, we introduce a bottleneck of information propagation between the sensory and motor networks by using a step-function nonlinearity on the output of the sensory network (Eq. 4). Similarly to the decision task of the static network, the output of the sensory network now becomes binary. This effectively reduces the flow of information from the sensory to the motor network, forcing the sensory network to consistently decide whether food should be consumed (with the caveat that the motor network can still interpret the binary sign in either of two ways, either consuming food marked with 1 or the ones marked with 0 by the sensory network). The agents perform equally well in this variation of the task as before (Fig. 5d), but now, the evolved plasticity rules seem to be more structured (Fig. 6b). Moreover, the variance of the learned weights in the best-performing agents is significantly reduced (Fig. 6d), which indicates that the bottleneck in the sensory network is increasing selection pressure for rules that learn the environment's food distribution accurately.

Discussion

We find that different sources of variability have a strong impact on the extent to which evolving agents will develop neuronal plasticity mechanisms for adapting to their environment. A diverse environment, a reliable sensory system, and a rate of environmental change that is neither too large nor too small are necessary conditions for an agent to be able to effectively adapt via synaptic plasticity. Additionally, we find that minor variations of the task an agent has to solve or the parametrization of the network can give rise to significantly different plasticity rules.

Our results partially extend to embodied artificial agents performing a foraging task. We show that environmental variability also pushes the development of plasticity in such agents. Still, in contrast to the static agents, we find that the interaction of a static motor network with a plastic sensory network gives rise to a much greater variety of well-functioning learning rules. We propose a potential cause of this degeneracy; as the relatively complex motor network is allowed to read out and process the outputs from the plastic network, any consistent information coming out of these outputs can be potentially interpreted in a behaviorally useful way. Reducing the information the motor network can extract from the sensory system significantly limits learning rule variability.

Our findings on the effect of environmental variability concur with the findings of previous studies (Lange and Sprekeler, 2020) that have identified the constraints that environmental variability places on the evolutionary viability of learning behaviors. We extend these findings in a mechanistic model which uses a biologically plausible learning mechanism (synaptic plasticity). We show how a simple evolutionary algorithm can optimize the different parameters of a simple reward-modulated plasticity rule for solving simple prediction and decision tasks. Reward-modulated plasticity has been extensively studied as a plausible mechanism for credit assignment in the brain (Florian, 2007; Baras and Meir, 2007; Legenstein et al., 2008) and has found several applications in artificial intelligence and robotics tasks (Burms et al., 2015; Bing et al., 2019). Here, we demonstrate how such rules can be very well-tuned to take into account different environmental parameters and produce optimal behavior in simple systems.

Additionally, we demonstrate how the co-evolution of plasticity and static functional connectivity in different subnetworks fundamentally changes the evolutionary pressures on the resulting plasticity rules, allowing for greater diversity in the form of the learning rule and the resulting learned connectivity. Several studies have demonstrated how, in biological networks, synaptic plasticity heavily interacts with (Butz et al., 2014; Stampanoni Bassi et al., 2019; Bernáez Timón et al., 2022) and is driven by network topology (Giannakakis et al., 2023). Moreover, it has been recently demonstrated that biological plasticity mechanisms are highly redundant in the sense that any observed neural connectivity or recorded activity can be achieved with a variety of distinct, unrelated learning rules (Ramesh, 2023). This observed redundancy of learning rules in biological settings complements our results and suggests that the function of plasticity rules cannot be studied independently of the connectivity and topology of the networks they are acting on.

The optimization of functional plasticity in neural networks is a promising research direction both as a means to understand biological learning processes and as a tool for building more autonomous artificial systems. Our results suggest that reward-modulated plasticity is highly adaptable to different environments and can be incorporated into larger systems that solve complex tasks.

Future work

This work studies a simplified toy model of neural network learning in stochastic environments. Future work could be built on this basic framework to examine more complex reward distributions and sources of environmental variability. Moreover, a greater degree of biological realism could be added by studying more plausible network architectures (possibly derived from connectomics data) and more sophisticated plasticity rule parametrizations.

Additionally, our foraging simulations were constrained by limited computational resources and were far from exhaustive. Further experiments can investigate environments with different constraints, food distributions, and multiple seasons as well as the inclusion of plasticity on the motor parts of the artificial organisms.

Acknowledgements

This work was supported by a Sofja Kovalevskaja Award from the Alexander von Humboldt Foundation. EG and SK thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for their support. We acknowledge the support from the BMBF through the Tübingen AI Center (FKZ: 01IS18039A). AL is a member of the Machine Learning Cluster of Excellence, EXC number 2064/1 – Project number 39072764.

References

Baras, D. and Meir, R. (2007). Reinforcement Learning, Spike-Time-Dependent Plasticity, and the BCM Rule. *Neural Computation*, 19(8):2245–2279.

Bernáez Timón, L., Ekelmans, P., Konrad, S., Nold, A., and Tchumatchenko, T. (2022). Synaptic plasticity controls the emergence of population-wide invariant representations in balanced network models. *Phys. Rev. Res.*, 4:013162.

Biesialska, M., Biesialska, K., and Costa-jussà , M. R. (2020). Continual lifelong learning in natural language processing: A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*. International Committee on Computational Linguistics.

Bing, Z., Baumann, I., Jiang, Z., Huang, K., Cai, C., and Knoll, A. (2019). Supervised learning in snn via reward-modulated spike-timing-dependent plasticity for a target reaching vehicle. *Frontiers in Neurorobotics*, 13.

Burms, J., Caluwaerts, K., and Dambre, J. (2015). Reward-modulated hebbian plasticity as leverage for partially embodied control in compliant robotics. *Frontiers in Neurorobotics*, 9.

Butz, M., Steenbuck, I., and van Ooyen, A. (2014). Homeostatic structural plasticity increases the efficiency of small-world networks. *Frontiers in Synaptic Neuroscience*, 6.

Caroni, P., Donato, F., and Muller, D. (2012). Structural plasticity upon learning: regulation and functions. *Nature reviews. Neuroscience*, 13(7):478—490.

Citri, A. and Malenka, R. (2008). Synaptic plasticity: Multiple forms, functions, and mechanisms. *Neuropsychopharmacology : official publication of the American College of Neuropsychopharmacology*, 33:18–41.

Confavreux, B., Zenke, F., Agnes, E., Lillicrap, T., and Vogels, T. (2020). A meta-learning approach to (re)discover plasticity rules that carve a desired function into a neural network. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, *Advances in Neural Information Processing Systems*, volume 33, pages 16398–16408. Curran Associates, Inc.

Deb, K. (2011). *Multi-objective Optimisation Using Evolutionary Algorithms: An Introduction*, pages 3–34. Springer London, London.

Dunlap, A. S. and Stephens, D. W. (2009). Components of change in the evolution of learning and unlearned preference. *Proceedings of the Royal Society B: Biological Sciences*, 276(1670):3201–3208.

Dunlap, A. S. and Stephens, D. W. (2016). Reliability, uncertainty, and costs in the evolution of animal learning. *Current Opinion in Behavioral Sciences*, 12:73–79. Behavioral ecology.

Ellefsen, K. O. (2014). The evolution of learning under environmental variability. pages 649–656.

Eskridge, B. E. and Hougen, D. F. (2012). Nurturing promotes the evolution of learning in uncertain environments. In *2012 IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL)*, pages 1–6.

Fawcett, T. W., Hamblin, S., and Giraldeau, L.-A. (2012). Exposing the behavioral gambit: the evolution of learning and decision rules. *Behavioral Ecology*, 24(1):2–11.

Feldman, D. E. (2009). Synaptic mechanisms for plasticity in neocortex. *Annual Review of Neuroscience*, 32(1):33–55. PMID: 19400721.

Florian, R. V. (2007). Reinforcement Learning Through Modulation of Spike-Timing-Dependent Synaptic Plasticity. *Neural Computation*, 19(6):1468–1502.

Giannakakis, E., Vinogradov, O., Buendia, V., and Levina, A. (2023). Recurrent connectivity structure controls the emergence of co-tuned excitation and inhibition. *bioRxiv*.

Guttenberg, N. (2019). Evolutionary rates of information gain and decay in fluctuating environments. ALIFE 2019: The 2019 Conference on Artificial Life:365–371.

Jordan, J., Schmidt, M., Senn, W., and Petrovici, M. A. (2021). Evolving interpretable plasticity for spiking networks. *eLife*, 10:e66273.

Kerr, B. and Feldman, M. (2003). Carving the cognitive niche: Optimal learning strategies in homogeneous and heterogeneous environments. *Journal of Theoretical Biology*, 220(2):169–188.

Khajehabdollahi, S., Prosi, J., Giannakakis, E., Martius, G., and Levina, A. (2022). When to Be Critical? Performance and Evolvability in Different Regimes of Neural Ising Agents. *Artificial Life*, 28(4):458–478.

Lange, R. T. and Sprekeler, H. (2020). Learning not to learn: Nature versus nurture in silico.

Lee, C. and Lee, A. (2020). Clinical applications of continual learning machine learning. *The Lancet Digital Health*, 2:e279–e281.

Legenstein, R., Pecevski, D., and Maass, W. (2008). A learning theory for reward-modulated spike-timing-dependent plasticity with application to biofeedback. *PLOS Computational Biology*, 4(10):1–27.

Magee, J. C. and Grienberger, C. (2020). Synaptic plasticity forms and functions. *Annual Review of Neuroscience*, 43(1):95–117. PMID: 32075520.

Najarro, E. and Risi, S. (2020). Meta-learning through hebbian plasticity in random networks. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, *Advances in Neural Information Processing Systems*, volume 33, pages 20719–20731. Curran Associates, Inc.

Nolfi, S. and Parisi, D. (1996). Learning to adapt to changing environments in evolving neural networks. *Adaptive Behavior*, 5(1):75–98.

Open Ended Learning Team, Stooke, A., Mahajan, A., Barros, C., Deck, C., Bauer, J., Sygnowski, J., Trebacz, M., Jaderberg, M., Mathieu, M., McAleese, N., Bradley-Schmieg, N., Wong, N., Porcel, N., Raileanu, R., Hughes-Fitt, S., Dalibard, V., and Czarnecki, W. M. (2021). Open-ended learning leads to generally capable agents.

Papini, M. R. (2012). *Evolution of Learning*, pages 1188–1192. Springer US, Boston, MA.

Parisi, G. I., Kemker, R., Part, J. L., Kanan, C., and Wermter, S. (2019). Continual lifelong learning with neural networks: A review.

Pedersen, J. W. and Risi, S. (2021). Evolving and merging hebbian learning rules: Increasing generalization by decreasing the number of rules. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '21, page 892–900, New York, NY, USA. Association for Computing Machinery.

Ramesh, P. (2023). *GANs schön kompliziert: Applications of Generative Adversarial Networks*. PhD thesis, University of Tübingen.

Schmidhuber, J. (1987). Evolutionary principles in self-referential learning. on learning now to learn: The meta-meta-meta...-hook. Diploma thesis, Technische Universitat Munchen, Germany.

Snell-Rood, E. C. (2013). An overview of the evolutionary causes and consequences of behavioural plasticity. *Animal Behaviour*, 85(5):1004–1011. Special Issue: Behavioural Plasticity and Evolution.

Snell-Rood, E. C. and Steck, M. K. (2019). Behaviour shapes environmental variation and selection on learning and plasticity: review of mechanisms and implications. *Animal Behaviour*, 147:147–156.

Stampanoni Bassi, M., Iezzi, E., Gilio, L., Centonze, D., and Buttari, F. (2019). Synaptic plasticity shapes brain connectivity: Implications for network topology. *International Journal of Molecular Sciences*, 20(24).

Thangarasa, V., Miconi, T., and Taylor, G. W. (2020). Enabling continual learning with differentiable hebbian plasticity.

Thornton, A. and Clutton-Brock, T. (2011). Social learning and the development of individual and group behaviour in mammal societies. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 366:978–87.

Yaman, A., Iacca, G., Mocanu, D. C., Coler, M., Fletcher, G., and Pechenizkiy, M. (2021). Evolving Plasticity for Autonomous Learning under Changing Environmental Conditions. *Evolutionary Computation*, 29(3):391–414.

Zenke, F. and Gerstner, W. (2017). Hebbian plasticity requires compensatory processes on multiple timescales. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 372(1715):20160259.