A Bio-Inspired Minimal Model for Non-Stationary K-Armed Bandits

Krubeal Danieli[1] and Mikkel Elle Lepperød[2]

[1]Center for Integrative Neuroplasticity, FYSCELL, University of Oslo, Norway
[2]Simula Research Laboratory, Oslo, Norway

Abstract

While reinforcement learning algorithms have made significant progress in solving multi-armed bandit problems, they often lack biological plausibility in architecture and dynamics. Here, we propose a bio-inspired neural model based on interacting populations of rate neurons, drawing inspiration from the orbitofrontal cortex and anterior cingulate cortex. Our model reports robust performance across various stochastic bandit problems, matching the effectiveness of standard algorithms such as Thompson Sampling and UCB. Notably, the model exhibits adaptive behavior: employing greedy strategies in low-uncertainty situations while increasing exploratory behavior as uncertainty rises. Through evolutionary optimization, the model's hyperparameters converged to values that align with known synaptic mechanisms, particularly in terms of synapse-dependent neural activity and learning rate adaptation. These findings suggest that biologically-inspired computational architectures can achieve competitive performance while providing insights into neural mechanisms of decision-making under uncertainty.

1 Introduction

The ability to make decisions for long-term reward maximization is a fundamental aspect of cognition. The brain has evolved specialized and interconnected regions to implement this behaviour under the constraints of biology.

Well-studied ecological settings of decision-making are foraging tasks, such as food search. In these problems, the agent is usually asked to choose between different options to maximize an expected reward. In nature, animals have been shown to exhibit different strategies depending on context. *Matching behaviour* is a well-known phenomenon in which the animal's decision patterns are proportional to the reward probability of the available options. Such behaviour is thought to result from the trade-off between exploration and exploitation [1, 2]. In fact, this is a well known phenomenon in the reinforcement learning literature, in which an agent is faced with the dilemma of exploring new alternatives, potentially more rewarding, or exploiting known options, despite being possibly sup-optimal.

A popular formalization of these type of tasks is the *multi-armed bandit* problem (MAB) [3]. This setting is usually described in terms of a slot machine endowed with K distinct arms, also called levers. During a round, the agent selects one of the arms and collects a reward R according to an unknown reward probability specific to the chosen arm. The goal is simply to maximize the total reward after a given number of steps, which is achieved by effectively updating a selection policy after each round. This problem has been extensively studied in the context of reinforcement learning, and it is considered a fundamental building block for more complex tasks [1].

The multi-armed bandit problem comes in several variants, with the simplest featuring a stationary reward distribution. Researchers have proposed numerous algorithms to address this problem, each with distinct theoretical guarantees. Thompson sampling, a widely adopted approach based on Bayesian optimization, maintains a posterior distribution over action reward probabilities and has demonstrated near-optimal regret bounds in stochastic settings [4, 5]. Similarly, the Upper Confidence Bound (UCB) algorithm achieves near-optimal regret bounds in adversarial settings by maintaining upper limits on action reward probabilities [6]. Other notable approaches include ϵ-Greedy and VDBE [7, 8, 9, 10]. However, these traditional algorithms, despite their effectiveness, lack biological plausibility – they neither resemble neural circuits nor capture brain dynamics. While not the primary driver, these limitations align with a growing intereset in machine learning towards bio-inspired algorithms, such as neural networks and predictive coding [11, 12], offering several advantages. Indeed, these methods can achieve state-of-the-art performance in various domains, including the challenging *machine-challenging tasks* (MCTs), set of problems that are hard for machines but relatively easy for humans [13, 14, 11]. Further, bio-inspired models enhance algorithmic interpretability by clarifying the functional relationships between internal components. When applied to tasks with existing experimental data, these models can generate new insights into brain dynamics and suggest novel research directions [15]. Finally, given the brain's remarkable energy efficiency, architectures inspired by its design may significantly reduce power consumption [16], particularly when implemented on specialized neuromorphic hardware [17].

In this work, we aimed at improving the biological plausibility of algorithms within multi-armed bandit tasks by introducing a novel model comprising two interacting rate neuron populations exhibiting synaptic plasticity. Our model addresses stochastic bandit problems with *concept drift* - scenarious where the reward distribution changes over time [18, 19, 20]. The proposed model achieves performance comparable to established algorithms like Thompson Sampling, ϵ-

Greedy and Upper Confidence Bound, while offering a more neurobiologically grounded interpretation of the decision-making process.

Architecturally, it is inspired by the prefrontal cortex, specifically modeling two critical regions: the orbitofrontal cortex (OFC) and anterior cingulate cortex (ACC). These brain areas are involved in motivational processing, expected value representation, action selection under uncertainty, and contextual interpretation [21, 22, 23, 24, 25]. The model's plasticity mechanism is based on a non-associative rule dependent on inter-population connection weight magnitudes. This approach aligns with synapse-type specific plasticity (STSP), a mechanism supported by emerging evidence linking learning rates to synaptic resource availability, current states, and morphological characteristics [26, 27, 28, 29]. Previous work has made use of such plasticity methods in spiking neural networks and synaptic metaplasticity models [30, 31].

Finally, the significance of our work is to bridge the gap between adaptive decision making under uncertainty and computational neuroscience. Our approach represents a possible way for how neural systems may implement choice selection in non-stationary environments, providing insights that could inform both artificial intelligence design and experimental research on adaptive behavior.

The remainder of this paper first describes our model design and learning, then presents experimental results and comparative analyses with established algorithms, lastly discusses the findings' broader implications and potential future directions.

2 Methods

The following section is organized as follows. First, we introduce a formalization of general problem setting, together with the variants considered in this work. Then, we outline the architecture of our model and how it can be mapped to neurobiology. Finally, we describe the learning procedure, and showcase its dynamics in a simple example.

2.1 Binomial K-armed bandit problem

The standard formulation of the task is structured as a set of K arms (or levers) $\mathcal{A}_K = \{a_1 \dots a_K\}$, with an associated reward distribution $\mathbf{p} = \{p_1, \dots p_K\}$. At each iteration, the agent pulls an arm and collects a possible reward drawn as a Bernoulli variable $R \sim \mathcal{B}(\{0, 1\}, p_k)$. The agent's objective is maximizing the total reward $\sum_t^T R_t$, after a certain number of rounds T, also called horizon. Importantly, the agent is unaware of the true reward probabilities, and thus has to make its decisions following a certain policy, denoted as π. In the reinforcement learning literature, the policy is often defined as a distribution over actions, here the arms $\mathcal{A}_K$, given the current state at time t. In the bandit problem, the state can be taken to correspond to the history h_t of past actions and rewards in the period $(0 \dots t]$, and the policy as a function that return a selected arm $\pi(h_t) = a_t$ [32].

Given the inherent stochasticity of the feedbacks from the environment, the policy is affected by the so-called exploration-exploitation trade-off, which here is phrased as the contrast between the option of the arm with the estimated highest expected reward versus the option to explore other arms, so to gather more information. A common approach is the $\epsilon-$greedy policy, where the choice to explore is selected with a probability ϵ. Moreover, it is often preferable to have a more explorative behaviour early during the training, with the intent to have a good sample size for the empirical reward distribution, which can be later exploited for maximizing reward.

Another important concept in multi-armed bandit problems is *regret.* Intuitively, it quantifies the loss of reward due to following a certain policy, and it is determined by the difference beween the collected reward and the theoretical optimal, obtained by choosing the best arm at each round. Formally, given defined a function $r(\pi)$ which returns the expected reward while following policy π, the regret ρ over an horizon T can be formulated as:

$$\rho = \frac{1}{T} \sum_{t}^{T} p_t^* - r(\pi(h_t)) \tag{1}$$

where p_t^* is the expected reward of the optimal arm at time t, which correspond to its probability since it is a Bernoulli distribution. The goal of the agent is to minimize the regret, and thus maximize the total reward.

2.2 Model description

The model is constructed as a rate network composed of two neuronal populations, U and V. The first, U, represents the memory traces of the K available options (*i.e.*, the bandits), while the second, V, encodes their values under the current policy. This architecture is inspired by the prefrontal cortex (PFC) and its central role in decision-making processes. Specifically, the two populations correspond to the orbitofrontal cortex (OFC) and the anterior cingulate cortex (ACC), respectively.

The OFC is known to represent different options, updating their values based on reward history and outcomes [33, 34]. Meanwhile, the ACC has been associated with action valuation and regulating the balance between exploration and exploitation [35]. Additionally, the dynamic interaction between the ACC and OFC has been linked to transient pre-stimulus activations, which bias decisions toward the most valuable option [36, 37, 38].

In our model, the first layer represents the available options, while learned connections to the second layer encode their values based on recent reward history. A key simplification is the lumping of option representations into single neurons. While this choice abstracts the more distributed encoding found in actual brain networks, it allows for a more tractable model design [39].

More formally, the model is defined by a set of coupled ordinary differential equations (ODEs). The first equation describes the evolution of the neural activity $\mathbf{u}$ in population U, while the second governs the activity $\mathbf{v}$ in population V, each evolving with its respective time constant τ.

$$\begin{aligned} \tau_u \dot{\mathbf{u}} &= -\mathbf{u} + \mathbf{W}^{VU} \phi_v(\mathbf{v}) + \mathbf{I}_{\text{ext}} \\ \tau_v \dot{\mathbf{v}} &= -\mathbf{v} + \widetilde{\mathbf{W}}^{UV} \phi_u(\mathbf{u}) \end{aligned} \tag{2}$$

The external input $\mathbf{I}_{\text{ext}}$ is a constant input that is used to set the initial conditions of the neural activity $\mathbf{u}$. The activation functions ϕ_v, ϕ_u are applied to population v and u respectively, and represent two distinct neural response functions tailored to each population vector. They have been chosen to be a step-function with threshold θ_v, θ_u applied to a generalized sigmoid with gain g_v, g_u and offset s_v, s_u.

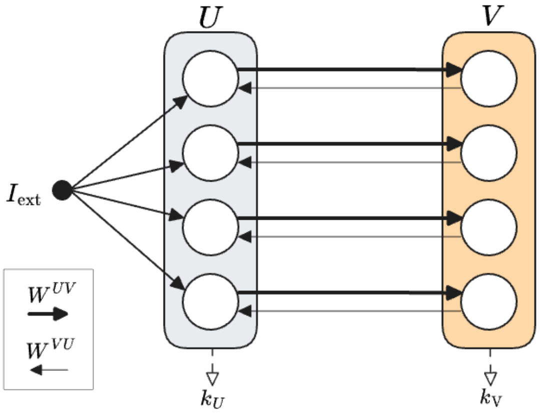

Figure 1: Model architecture - *The model is composed of a layer U (blue), receiving a feedfoward input I_{ext}, a layer V (orange), and connections $\boldsymbol{W}^{UV}$ and $\boldsymbol{W}^{VU}$. Additionally, two indexes k_U, k_V are extracted from the layers and corresponds to the selection made by the two populations as $k_U = argmax_k\{\boldsymbol{u}\}$, $k_V = argmax_k\{\boldsymbol{v}\}$.*

Importantly, the two layers are not fully connected and the matrices are diagonal. More in detail, the weight matrix $\mathbf{W}^{VU}$ is simply made of 1s, while $\widetilde{\mathbf{W}}^{UV}$ is a function of the actual weights $\Phi_v(\mathbf{W}^{UV})$ and it represents the contribution of the active options $\mathbf{u}$ to the value representation $\mathbf{v}$, it is thus referred to as *option value function*. The matrix $\mathbf{W}^{UV}$ is initialized to all zeroes. The function Φ_v is defined as weighted sum of a generalized sigmoid and a Gaussian, whose shape is characterized by a bell curve smoothly settling to a constant value. For details see the appendix 5.

The motivation behind our choice of Φ_v is to be agnostic about its final form, and allow competition or integration of two distinct traits of the function shape. In particular, one corresponds to a smooth transition to a plateau value with a certain steepness (or gain), which can represent a saturation once a threshold is crossed, such features has been reported for both biological and artificial neurons [40, 41]. The other is a bell-shaped curve with a defined center and width, which can allow for placing emphasis on values only within a given window and modulate information transfer [42].

The model hyperparameters were optimized for maximizing the average total reward over multiple runs. In particular, given the non-differentiability of the model with respect to the fitness function we employed an evolutionary algorithm, more specifically CMA-ES.

2.2.1 Option selection

The decision-making process within a single round is structured in two distinct phases. Initially, the model receives a constant external input targeting all neurons in the memory population U equally. During this phase, $\mathbf{I}_{\text{ext}}$ works as an equilibrium value while the reciprocal interactions with population V push $\mathbf{u}$ to different values, depending on the current policy encoded in $\widetilde{\mathbf{W}}^{UV}$. Importantly, the weights $\mathbf{W}^{UV}$ are initialized to zero, and thus the input from U to V is uniform. This approach ensures the absence of biases towards any arm by having all weights equal, and corresponds to a completely untrained network. After a fixed amount of time $\sim$ 2s, the second phase begins. Here, the external input is removed and the model is left to evolve autonomously, and since there are no recurrent connections in neither population the dynamics are entirely driven by their coupling. A selection k is sampled after another fixed amount of time $\sim$ 5s, and it is defined according to the following rule:

$$k = \begin{cases} \text{argmax}_k\{\mathbf{v}\} & \textit{if } \text{argmax}_k\{\mathbf{v}\} = \text{argmax}_k\{\mathbf{u}\} \\ \text{random}(K) & \textit{otherwise} \end{cases}$$

The selection rule is simple: if the value representation $\mathbf{v}$ is in agreement with the memory trace $\mathbf{u}$, then the option with the highest value is selected. Otherwise, a random option is chosen. This rule is a way to express the exploration-exploitation trade-off, and it is dependent on the current value of the weights $\widetilde{\mathbf{W}}^{UV}$.

Below in subsection 2.2.1, it is reported the pseudo-code for the algorithm behind the selection process, which is applied during each round t.

Lastly, the structure of the option selection process resembles the prefrontal circuitry, as choices emerge from network state sampling following a period of autonomous neural activity. The stability of these neural activations depends on the strength and reliability of the highest-valued option [43, 44].

According to the values of the policy's parameters, the behaviour of the model displays periods of exploration followed by a steady exploitation, which can be reverted in case of a change in the environment's reward distribution.

Algorithm 1: Two-phases option selection process

```
Input: External input $\mathbf{I}_{\text{ext}}$, population $\mathbf{u}$, population $\mathbf{v}$, weights $\widetilde{\mathbf{W}}^{UV}$
Output: Selected action $k$
Phase 1: external input ;                              // Duration:  ~2s
Define constant $\mathbf{I}_{\text{ext}}$;
Update populations $\mathbf{u}, \mathbf{v}$ according to 2.2;
Phase 2: autonomous evolution ;                        // Duration:  ~2s
Remove external input $\mathbf{I}_{\text{ext}}$;
Let system evolve through population coupling according to 2.2;
Selection process:;
$k_u \leftarrow \text{argmax}_k\{\mathbf{u}\}$;
$k_v \leftarrow \text{argmax}_k\{\mathbf{v}\}$;
if $k_u = k_v$ then
    $k \leftarrow k_v$ ;                                  // Exploitation
else
    $k \leftarrow \text{random}(K)$ ;                     // Exploration
end
return $k$
```

2.3 Learning

Given a selected option k, the environment (set of bandits) samples and returns a reward $R \in \{0, 1\}$ with probability p_k. Then, the weights $\mathbf{W}^{UV}$ for the neuron corresponding to the option k are updated according to the following plasticity rule:

$$\Delta \mathbf{W}_k^{UV} = \tilde{\eta}_k \left(R \cdot w^+ - \mathbf{W}_k^{UV} \right) \tag{3}$$

where w^+ is a constant maximum synaptic weight, while $\tilde{\eta}_k$ is the learning rate for the option k determined by a function Φ_η of the current weights $\mathbf{W}_k^{UV}$, referred to as *learning rate function.*

The shape of Φ_η is again a Gaussian-sigmoid but with different parameters, giving evolution the opportunity to combine the two characteristic traits of plateau and bell-shaped tuning. In particular, these features can be combined so to define mechanisms of synapse-type specific plasticity as a function of the current synaptic strenght [26], as well the application of other useful homeostatic constraints with computational advantages, such as synaptic scaling and proportional updates [45, 46, 47].

3 Experiments

The model has been tested in a series of benchmark environments, each with a different number of arms and reward distributions. The performance has been compared with the following algorithms: Random Baseline, Upper-Confidence Bound (UCB), Thompson Sampling, and Epsilon-Greedy.

3.1 Game variants

Our goal is to investigate the performance of the agent in a non-stationary environment, meaning that its underlying distribution changes over time [1] We choose this setting as it resembles an scenario in which an animal has to forage in an environment with food (reward) distributed over a set of fixed locations, but whose occurrence probability can change over time. Four different variants were used, obtained by introducing different types of non-stationarity: piecewise constant, uniformly changing, sinusoidally changing, and sinusoidally changing with piecewise constant arms. The reason for these choices is to test the model performance under different speed and uniformity of the distribution changes. Figure 2 visually illustrates their specificities.

Piecewise stationary environment [KAB-P]
Within a trial the reward distribution is stationary and it is drawn from a normal $\mathbf{p} = \mathcal{N}(0.5, 0.2)^K$, clipped in $(0, 1)$. At the end of each trial i it is drawn a new distribution $\mathbf{p}_i \rightarrow \mathbf{p}_{i+1}$ [32].

Piecewise stationary environment with drift [KAB-D]
At the very beginning, the reward distribution $\mathbf{p}$ is sampled from a normal $\mathbf{p} = \mathcal{N}(0.5, 0.2)^K$. Then, it changes gradually over the rounds, tracked as time t, such that its values tend towards a target distribution $\mathbf{q}_i$ as $\tau_p \dot{\mathbf{p}}_t = \mathbf{q}_i - \mathbf{p}_t$. Here, $\dot{\mathbf{p}}$ is the time derivative of the distribution and τ_p is its time constant. Once the distance is below a threshold δ as $|\mathbf{q}_i - \mathbf{p}_t| < \delta$, the target distribution is changed to a new one $\mathbf{q}_i \rightarrow \mathbf{q}_{i+1}$. In this variant, there are no proper trials but the target distribution keep changing until a maximum number of rounds is reached.

Sinusoidal distribution shift [KAB-sin]
The reward distribution changes over rounds, with the probability of each arm following a sine wave with a specific frequency f_k, phase λ_k and amplitude 1. At any given time t, the distribution is $\mathbf{p}_t = \{\sin(2\pi f_k t + \lambda_k)$ for $k = 1 \ldots K\}$.

Partial sinusoidal distribution shift [KAB-sinP]
Identical to the sinusoidal distribution shift, but only a subset of the arms changes sinusoidally while the rest is kept at a constant value and the distribution is not normalized.

[1] Since the arm probabilities are not normalized to 1, it is technically improper to call them *probability distributions*; we will therefore refer to either *probability* or *distribution* separately at any given time for avoiding confusion.

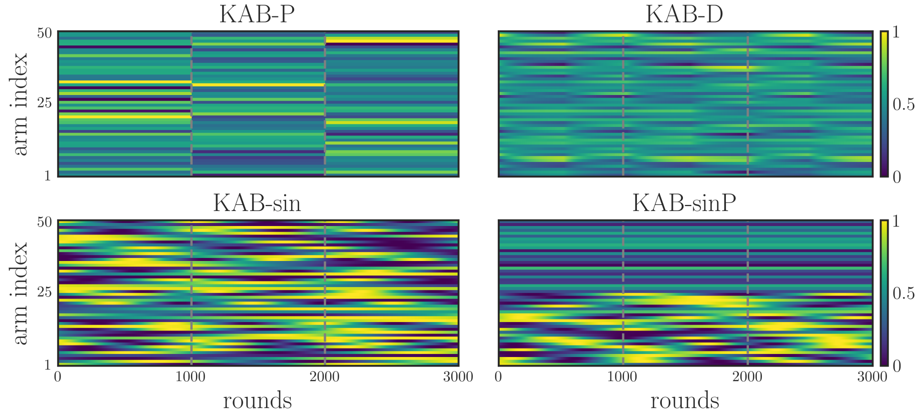

Figure 2: Reward distribution for the four game variants - *The reward distribution for each arm and environment is plotted over three trials of 1000, demarcated by a dotted grey line.*

3.2 Evolution search

The optimization of the hyper-parameters was performed using the Covariance Matrix Adaptation evolutionary strategy algorithm (CMA-ES) [48]. The search was run with a population of 256 individuals for 80 generations. Each individual was endowed with a genome, corresponding to a vector of 22 parameters of the model. The fitness function of the evolution was defined as the average reward obtained by an individual over 3 different non-stationary bandit environments, each for $K = \{40, 200\}$, and all averaged over 2 iterations. The results are summarized below in figure 3.

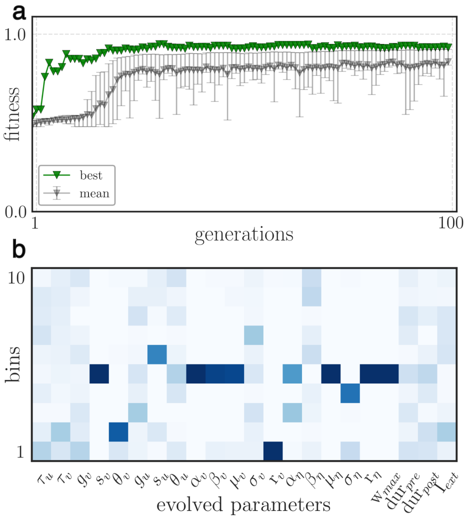

Figure 3: Evolution results - **a**: *top fitness, mean and standard deviation (as 16-84 percentile) of the population over generations.* - **b**: *heatmap of the evolved parameters (rows) as histogram bins (y-axis) calculated from the 50 percentile of the population of the last generation; higher density is in dark blue* - **c-d**: *Gaussian-sigmoid [c] and neural response functions [d] of the top-half of the population, the color intensity is proportional to the fitness.*

The evolution progress, plot **3a**, showed a steady improvement over the generations, with the fitness hitting a plateau once the total reward could not get any closer to the optimum ≈ 1 due to randomness and the finite horizon.

The evolved parameters (genome) are displayed as a frequency heatmap in **3b**, which was obtained from the individuals whose fitness was equal or above the median. The most relevant parameters resulted to be those concerning directly neuronal activity and learning.

More in details, the two Gaussian-sigmoid functions Φ_v, Φ_η, which are respectively the option value and learning rate functions, are shown in **3c**. On the one hand, the learning rate function Φ_η is characterized by a decreasing curve, associated with the dominance of the right-side of the Gaussian, and a completely ignored sigmoid component. A possible interpretation for this shape is to ensure a high learning speed when the value options are more uncertain (weak weights), and low otherwise; thus preventing overshooting and oscillations in the weight updates. This adaptive behaviour is in line with known neuronal dynamics such as homeostatic plasticity, which works towards a stabilization of synapses, for instance through synaptic scaling and proportional updates [45]. A variable learning rate is an important feature of several plasticity rules, from the more biologically plausible like the Oja rule [49] to deep learning optimizers like Adam [50].

On the other hand, the option value function Φ_v follows instead a steep sigmoid curve. This is consistent with the idea that the input of population U to population V is weighted maximally for high option values (strong synapses), whereas for weaker estimates the contributions are low or close to zero, allowing for more exploration. Interestingly, a common feature seemed to be a slight concavity after zero, a slim influence of the Gaussian component, which might be interpreted as a sort of test for newly formed synapses. However, the size of this effect is not large.

The neural response functions are shown in **3d**. Both population evolved to have a similar shape, a sharp sigmoid with a clear threshold, with population U having a more variable distribution. The form is characterized by not allowing for a fine-grained linear reponse but rather an high-pass filter, with activity occurring only after strong excitation. This firing behaviour is reminiscent of coincidence detector neurons, which are sometimes referred to as class III neurons with respect to their f-I curve [51].

3.3 Environment variants and number of arms

The model has been tested and compared with the other algorithms: Thompson Sampling (TS), ϵ-Greedy, and UCB. The benchmark were the four different variants of the K-armed bandit problem listed above 3.1, with a variable number of arms ranging from 5 to 1000. The results are reported in table 3.3. Overall, our model displays a solid performance over all environments, most of the time being equally good or better than the other algorithms. Interestingly, a large numbers of arms (K) did not present a significant challenge, as the model effectively adapted to the different environments and reward distributions. However, this is in part due to the randomness in the assignment of arm probabilities, and the statistics of the quantity of high-reward arms as their number increases. Nonetheless, given the non-stationarity it is still a non trivial task to re-calibrate to new distributions.

3.4 Analysis of dynamics and robustness

3.4.1 Entropy analysis

For a better understanding of the qualitative differences between the models, we analyzed the progress over the rounds by tracking the selected arms in a simple piecewise stationary distribution environment. The simulation was run for 3 trials with 2000 rounds each, and then averaged over 5 iterations. Additionally, in order to quantify the variability of the decision policy at a given time and highlight the particularity of each decision-making behaviour, we calculated the entropy of the probability distribution p of chosen arms, calculated over a window of 20 rounds, as $H = -\sum_i^K p_i \log(p_i)$. The unit of entropy is in nats, and it ranges from 0 (no uncertainty) to $\log_e(K)$ (maximum uncertainty). In figure 4-**a**, it is plotted for each model the raster plot of selected arms together with its level of entropy. The reward probability distribution over the arms has an average of $H = 2.02$.

K	5	10	50	100	200	1000
TS	0.03(8)	0.02(6)	0.02(7)	0.04(5)	0.02(3)	0.16(2)
ϵ-Greedy	0.05(14)	0.07(5)	0.08(7)	0.15(6)	0.08(2)	0.10(4)
UCB	0.05(15)	0.05(8)	0.19(6)	0.33(3)	0.39(2)	0.54(3)
Model	**0.08(13)**	**0.07(11)**	**0.07(14)**	**0.07(8)**	**0.09(9)**	**0.07(8)**
TS	0.03(6)	0.08(13)	0.16(6)	0.19(3)	0.28(7)	0.34(3)
ϵ-Greedy	0.04(7)	0.14(13)	0.22(5)	0.19(8)	0.26(7)	0.16(4)
UCB	0.05(6)	0.09(13)	0.21(3)	0.36(4)	0.40(3)	0.49(2)
Model	**0.13(10)**	**0.15(16)**	**0.05(6)**	**0.21(5)**	**0.26(7)**	**0.12(7)**
TS	0.21(22)	0.22(16)	0.07(5)	0.10(5)	0.06(4)	0.21(5)
ϵ-Greedy	0.21(21)	0.18(10)	0.12(5)	0.12(6)	0.10(4)	0.10(1)
UCB	0.03(4)	0.05(3)	0.17(4)	0.23(1)	0.33(3)	0.49(3)
Model	**0.00(3)**	**0.02(4)**	**0.05(3)**	**0.06(4)**	**0.08(1)**	**0.05(4)**
TS	0.19(21)	0.43(19)	0.17(10)	0.11(6)	0.09(6)	0.19(6)
ϵ-Greedy	0.24(26)	0.43(10)	0.24(10)	0.14(5)	0.15(6)	0.14(2)
UCB	0.00(6)	0.26(17)	0.18(6)	0.29(3)	0.34(1)	0.52(3)
Model	**0.00(9)**	**0.23(17)**	**0.14(7)**	**0.08(8)**	**0.06(4)**	**0.09(7)**

Table 1: Table of performance - *From the top: results for MAB-P, MAB-D, MAB*-sin*, MAB*-sin*P for different numbers K of arms. Average regret and standard deviation (2 decimal places) over 2 trials of 2000 rounds each averaged over 5 simulations.*

As expected, the shape of the entropy curve expresses the inherent strategy adopted by each model. In particular, the UCB algorithm showed the highest variability, marked by a persistent exploratory behaviour throughout the trials despite converging to reward options. Thompson Sampling was able to reach most solutions, although with difficulty in adapting to new reward distributions leading to high entropy levels. $\epsilon-$Greedy also showed a good performance quite reliably, with the greedy strategy assuring low entropy for most of the rounds. Similar behaviour was observed for our model, which was able to reach the optimal policy and maintain it over time, with entropy peaking mostly at the beginning of the trials and being, on average, the lowest among all models. Indeed, the dynamics of our model make it particularly suited for the task of non-stationary K-armed bandits, as it is able to quickly adapt to new reward distributions and firmly maintain a greedy policy.

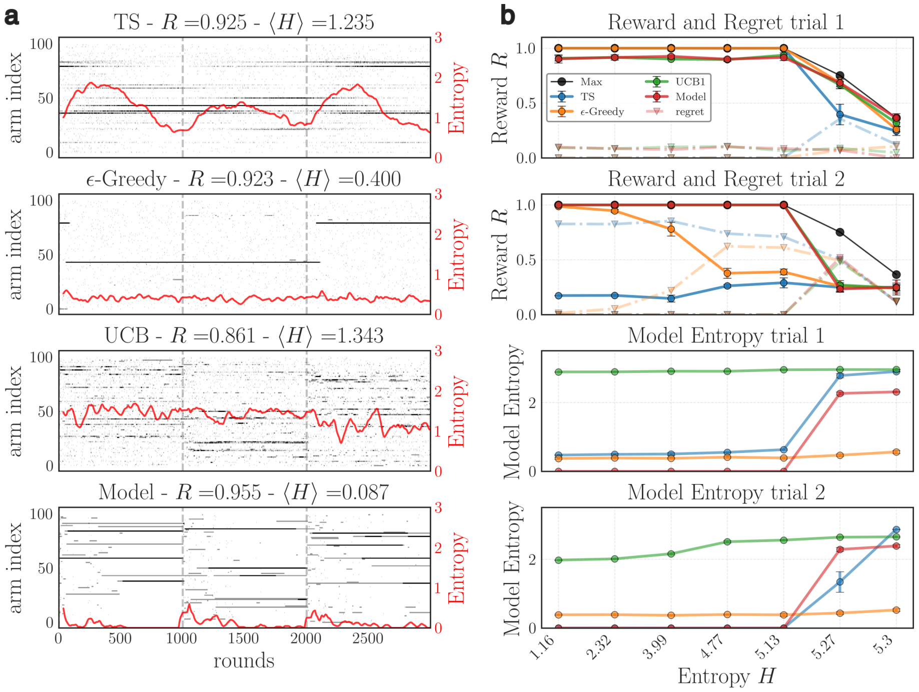

Figure 4: Decision-making dynamics for different models - **a**: *Each plot display the results from one model. The raster plots (black dots) show the arms selected at each round. The red lines represent the entropy level, calculated from the distribution of selections over the preceeding 20 rounds, smoothed with a 30-steps moving average. In the plot titles, the total reward and average entropy over all trials are also reported.* - **b**: *the top two rows display the average reward for trial 1 and 2 obtained by each model for increasing levels of entropy (in nats) in the reward distribution; a dashed line is the regret with respect to the upper bound (black solid line). The two bottom average entropy of the selections for the first and second trial of the simulation, each with 2000 rounds.*

Then, we sought to investigate the robustness of the model, quantified as the capacity to endure increasing levels of entropy in the reward distribution. The simulation was done in a piecewise stationary environment with $K = 200$ in two trials averaged over 5 independent runs, and it is showed in figure **4b**. The distributions were chosen such to have only one strongly rewarding arm, in order to highlight the models ability to find it. For more details about the distribution see the appendix 5.4. In the top two plots, it is shown the average reward and regret obtained by each model against the reward distribution entropy for the two trials. The results reported how all models are capable of robust performance in the first trial even in the presence of high uncertainty. In the second trial, ϵ-Greedy and Thompson Sampling suffered the increasing difficulty of switching arms, probably due to their conservative approaches. However, this challenge afflicted UCB and our model only with higher entropy levels, recognizing their adaptability.

Another perspective to this analysis was given by the two bottom plots, which showed the average entropy over the trials. Overall, there was the unsuprising trend of increasing selection entropy with the entropy of the reward distribution. Nonetheless, striking is the exception of Epsilon-Greedy, which still maintained a constant level throughout. UCB displayed the highest average values, while Thompson Sampling followed with some delay. On the other hand, our model display a more abrupt change, going from a state of very low to high variability, sign of a solid exploratory behaviour.

4 Discussion

The process of making decision in uncertain situations is a remarkable aspect of cognition. For instance, such behaviour is implemented in animals during foraging and matching behaviour. In the context of humans, it has been observed that the pool of adopted policies vary considerably [52]. Nevertheless, the human subjects seems able to integrate environmental uncertainty and trial generalization in their strategy, and Bayesian algorithms are generally a good fit for the observed policies [53, 54]. A useful formalization of such tasks are multi-armed bandit problems (MABs), which has been extensively studied in the context of reinforcement learning [1]. Although several algorithms have been proposed to solve the problem with robust theoretical guarentees, there is a general lack of biological plausibility of the architecture and dynamics.

The goal of this work was to design a bio-inspired architecture and learning for solving MABs. In particular, we introduced a model based on two interactive population of rate neurons to address the binomial K-armed bandit problem in non-stationary environments. We took inspiration from the functional role of the orbitofrontal cortex (OFC) and anterior cingulate cortex (ACC), two important pre-frontal regions known to be involved in decision-making processes [34, 35]. The results obtained report its adaptability to changing reward distributions and maintaining a rewarding policy over time, achieving equally well when compared to standard algorithms. The assessment was done over four different variants of stochastic bandit problems and a wide range of number of arms, providing evidence for the consistency of the model.

Further analysis involved the evaluation of the its behaviour in situations with variable levels of entropy in the reward distribution. One notable insight was that in situation with low uncertainty, the model was reliably capable of quickly switching to the rewarding option and settling to a greedy strategy, similarly to Thompson Sampling but unlike UCB, which is used to persevere in a noticeable exploratory behaviour. When the uncertainty increased over a certain level the option entropy of the model followed, which however did not necessarily hinder performance, except for switching arm in new trials. Here, the adopted policy became markedly exploratory, akin to the approach of UCB.

The strengths of the model can be traced both in the architecture and in the learning paradigm, whose hyperparameters were optimized through an evolutionary process. Interestingly, the values found converged to solutions that can be mapped to plausible synaptic mechanisms. On one hand the neural dynamics, which rely on plastic connections and a consensus-like selection process. Particularly important was the choice of modulating the afferent connections to the value population V according to a non-linear function dependant on the synaptic weight itself. In so doing, it was possible to evolve implicitly an effective option-value policy for the tradeoff between exploration and exploitation. This approach can be seen as a form of meta-plasticity implemented through neuromodulation [55], where a region external to the network affects the synaptic connections withough alterning their actual weights; dopamine is a well-suited candidate [56, 57, 58]. The emerged neural response functions were characterized by a steep sigmoidal shape, which can be related to the saturation of the neural response once a certain threshold is crossed, a feature observed in biological network as class III neurons, besides being a common choice for artificial ones [51, 40, 41].

On another hand, learning was structured as a non-associative plasticity rule based on the reward. Similarly to before, a non-linear function of the synaptic weights played a critical role, specifically in defining the synapse-specific learning rate [26]. Again, this mechanism can be considered a form of meta-learning, with evolution leading to the emergence of hyper-parameter encoding important inductive biases [30, 31]. Further, the evolved shape of the learning rate function was inversely proportional to the synaptic weight, which can be related to the resource availability at the synapse and its state, including the size [27, 28, 29].

Despite the promising results, there are some limitations to the model. First and foremost, the great level of abstraction in the neuronal details, as we considered simple point neurons with synapses modeled with relatively elementary functions, lacking the anatomical complexity of actual dendrites. In particular, the model does not account for the presence of noise in the neural dynamics, which is a well-known feature of biological neurons [59]. Further, the functional association with the pre-frontal cortical region is only moderate, although present. On the computational side, since our interest lied in the biological plausibility and evolution of adaptive meta-learning solutions, we used as reference only a few well established and relatively simple algorithms, and did not take into account more advanced variants such as VDBE [9, 10], *f-Discounted-Sliding-Window Thompson Sampling* (*f-dsw TS*) [20], and variants of ϵ-Greedy [32].

Future work could involve the comparison with more complex algorithms, and the introduction of more realistic neural dynamics, such as spiking neurons [60].

Acknowledgements & Statements

The authors declare no competing interests.

The code is publicly available and can be found at `https://github.com/`

iKiru-hub/minBandit.git.

This research was funded by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement N$^{\text{o}}$ 945371 and the University of Oslo.

The research presented in this paper has benefited from the Experimental Infrastructure for Exploration of Exascale Computing (eX3), which is financially supported by the Research Council of Norway under contract 270053.

Lastly, special thanks to Kosio Beshkov and Marianne Fyhn for inputs and feedback.

References

[1] Richard S. Sutton and Andrew G. Barto. The Reinforcement Learning Problem. In *Reinforcement Learning: An Introduction*, pages 51–85. MIT Press, 1998.

[2] Yael Niv, Daphna Joel, Isaac Meilijson, and Eytan Ruppin. Evolution of Reinforcement Learning in Uncertain Environments: A Simple Explanation for Complex Foraging Behaviors. *International Society for Adaptive Behavior*, 2002.

[3] Bruno B. Averbeck. Theory of Choice in Bandit, Information Sampling and Foraging Tasks. *PLoS Computational Biology*, 11(3):e1004164, March 2015.

[4] Shipra Agrawal and Navin Goyal. Analysis of Thompson Sampling for the Multi-armed Bandit Problem. In *Proceedings of the 25th Annual Conference on Learning Theory*, pages 39.1–39.26. JMLR Workshop and Conference Proceedings, June 2012.

[5] Emilie Kaufmann, Nathaniel Korda, and Rémi Munos. Thompson Sampling: An Asymptotically Optimal Finite Time Analysis, July 2012.

[6] Peter Auer and Nicolo Cesa-Bianchi. Finite-time Analysis of the Multi-armed Bandit Problem. *Machine Learning*, 2002.

[7] J. C. Gittins. Bandit Processes and Dynamic Allocation Indices. *Journal of the Royal Statistical Society. Series B (Methodological)*, 41(2):148–177, 1979.

[8] Yikun Ban, Jingrui He, and Curtiss B. Cook. Multi-facet Contextual Bandits: A Neural Network Perspective, June 2021.

[9] Michel Tokic. Adaptive ε-Greedy Exploration in Reinforcement Learning Based on Value Differences. In Rüdiger Dillmann, Jürgen Beyerer, Uwe D. Hanebeck, and Tanja Schultz, editors, *KI 2010: Advances in Artificial Intelligence*, volume 6359, pages 203–210. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[10] Michel Tokic and Günther Palm. Value-Difference Based Exploration: Adaptive Control between Epsilon-Greedy and Softmax. In Joscha Bach and Stefan Edelkamp, editors, *KI 2011: Advances in Artificial Intelligence*, volume 7006, pages 335–346. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[11] Jangho Lee, Jeonghee Jo, Byounghwa Lee, Jung-Hoon Lee, and Sungroh Yoon. Brain-inspired Predictive Coding Improves the Performance of Machine Challenging Tasks. *Frontiers in Computational Neuroscience*, 16:1062678, 2022.

[12] M. W. Spratling. A review of predictive coding algorithms. *Brain and Cognition*, 112:92–97, March 2017.

[13] Samuel Schmidgall, Rojin Ziaei, Jascha Achterberg, Louis Kirsch, S. Pardis Hajiseyedrazi, and Jason Eshraghian. Brain-inspired learning in artificial neural networks: A review. *APL Machine Learning*, 2(2):021501, May 2024.

[14] Demis Hassabis, Dharshan Kumaran, Christopher Summerfield, and Matthew Botvinick. Neuroscience-Inspired Artificial Intelligence. *Neuron*, 95(2):245–258, July 2017.

[15] Ziming Liu, Eric Gan, and Max Tegmark. Seeing is Believing: Brain-Inspired Modular Training for Mechanistic Interpretability, June 2023.

[16] Evaluation of Bio-Inspired Models under Different Learning Settings For Energy Efficiency in Network Traffic Prediction. https://arxiv.org/html/2412.17565.

[17] A Review of Neuroscience-Inspired Machine Learning. https://arxiv.org/html/2403.18929v1.

[18] Aurélien Garivier and Eric Moulines. On Upper-Confidence Bound Policies for Non-Stationary Bandit Problems, May 2008.

[19] Omar Besbes, Yonatan Gur, and Assaf Zeevi. Stochastic Multi-Armed-Bandit Problem with Non-stationary Rewards. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[20] Emanuele Cavenaghi, Gabriele Sottocornola, Fabio Stella, and Markus Zanker. Non Stationary Multi-Armed Bandit: Empirical Evaluation of a New Concept Drift-Aware Algorithm. *Entropy*, 23(3):380, March 2021.

[21] J. O'Doherty, M. L. Kringelbach, E. T. Rolls, J. Hornak, and C. Andrews. Abstract reward and punishment representations in the human orbitofrontal cortex. *Nature Neuroscience*, 4(1):95–102, January 2001.

[22] Justin S. Riceberg and Matthew L. Shapiro. Reward Stability Determines the Contribution of Orbitofrontal Cortex to Adaptive Behavior. *Journal of Neuroscience*, 32(46):16402–16409, November 2012.

[23] Léon Tremblay and Wolfram Schultz. Relative reward preference in primate orbitofrontal cortex. *Nature*, 398(6729):704–708, April 1999.

[24] Rebecca Elliott, Raymond J. Dolan, and Chris D. Frith. Dissociable Functions in the Medial and Lateral Orbitofrontal Cortex: Evidence from Human Neuroimaging Studies. *Cerebral Cortex*, 10(3):308–317, March 2000.

[25] Michael J. Frank and Eric D. Claus. Anatomy of a decision: Striato-orbitofrontal interactions in reinforcement learning, decision making, and reversal. *Psychological Review*, 113(2):300–326, April 2006.

[26] Rylan S Larsen and P Jesper Sjöström. Synapse-type-specific plasticity in local circuits. *Current opinion in neurobiology*, 35:127–135, December 2015.

[27] Arne V. Blackman, Therese Abrahamsson, Rui Ponte Costa, Txomin Lalanne, and P. Jesper Sjöström. Target-cell-specific short-term plasticity in local circuits. *Frontiers in Synaptic Neuroscience*, 5:11, December 2013.

[28] Thomas M. Bartol, Cailey Bromer, Justin Kinney, Michael A. Chirillo, Jennifer N. Bourne, Kristen M. Harris, and Terrence J. Sejnowski. Hippocampal Spine Head Sizes Are Highly Precise, March 2015.

[29] Pablo Ariel, Michael B. Hoppa, and Timothy A. Ryan. Intrinsic variability in Pv, RRP size, Ca(2+) channel repertoire, and presynaptic potentiation in individual synaptic boutons. *Frontiers in Synaptic Neuroscience*, 4:9, 2012.

[30] Jeffrey B. Inglis, Vivian V. Valentin, and F. Gregory Ashby. Modulation of Dopamine for Adaptive Learning: A Neurocomputational Model. *Computational brain & behavior*, 4(1):34–52, March 2021.

[31] Kiyohito Iigaya. Adaptive learning and decision-making under uncertainty by metaplastic synapses guided by a surprise detection system. *eLife*, 5:e18073, August 2016.

[32] Han Qi, Fei Guo, and Li Zhu. Forced Exploration in Bandit Problems, December 2023.

[33] Chung-Hay Luk and Jonathan D. Wallis. Choice Coding in Frontal Cortex during Stimulus-Guided or Action-Guided Decision-Making. *Journal of Neuroscience*, 33(5):1864–1871, January 2013.

[34] Steven W. Kennerley and Mark E. Walton. Decision Making and Reward in Frontal Cortex. *Behavioral Neuroscience*, 125(3):297–317, June 2011.

[35] Mehdi Khamassi, Pierre Enel, Peter Ford Dominey, and Emmanuel Procyk. Chapter 22 - Medial prefrontal cortex and the adaptive regulation of reinforcement learning parameters. In V. S. Chandrasekhar Pammi and

Narayanan Srinivasan, editors, *Progress in Brain Research*, volume 202 of *Decision Making*, pages 441–464. Elsevier, January 2013.

[36] Shintaro Funahashi. Prefrontal Contribution to Decision-Making under Free-Choice Conditions. *Frontiers in Neuroscience*, 11, July 2017.

[37] Encarni Marcos and Aldo Genovesio. Determining Monkey Free Choice Long before the Choice Is Made: The Principal Role of Prefrontal Neurons Involved in Both Decision and Motor Processes. *Frontiers in Neural Circuits*, 10, September 2016.

[38] Zuzanna Z. Balewski, Thomas W. Elston, Eric B. Knudsen, and Joni D. Wallis. Value dynamics affect choice preparation during decision-making. *Nature neuroscience*, 26(9):1575–1583, September 2023.

[39] Alex Martin. The Representation of Object Concepts in the Brain. *Annual Review of Psychology*, 58(1):25–45, January 2007.

[40] Gabriel Koch Ocker and Michael A. Buice. Flexible neural connectivity under constraints on total connection strength, January 2020.

[41] Andrea Apicella, Francesco Donnarumma, Francesco Isgrò, and Roberto Prevete. A survey on modern trainable activation functions. *Neural Networks*, 138:14–32, June 2021.

[42] Paul Miller and Jonathan Cannon. Combined mechanisms of neural firing rate homeostasis. *Biological Cybernetics*, 113(1):47–59, 2019.

[43] Lars Bäckman, Lars Nyberg, Anna Soveri, Jarkko Johansson, Micael Andersson, Erika Dahlin, Anna S. Neely, Jere Virta, Matti Laine, and Juha O. Rinne. Effects of Working-Memory Training on Striatal Dopamine Release. *Science*, 333(6043):718–718, August 2011.

[44] Pierre Enel, Joni D Wallis, and Erin L Rich. Stable and dynamic representations of value in the prefrontal cortex. *eLife*, 9:e54313, July 2020.

[45] Ami Citri and Robert C. Malenka. Synaptic Plasticity: Multiple Forms, Functions, and Mechanisms. *Neuropsychopharmacology*, 33(1):18–41, January 2008.

[46] Mary B. Kennedy. Synaptic Signaling in Learning and Memory. *Cold Spring Harbor Perspectives in Biology*, 8(2):a016824, February 2016.

[47] Mohammad Samavat, Thomas M. Bartol, Kristen M. Harris, and Terrence J. Sejnowski. Synaptic Information Storage Capacity Measured With Information Theory. *Neural Computation*, 36(5):781–802, April 2024.

[48] Christian Igel, Nikolaus Hansen, and Stefan Roth. Covariance Matrix Adaptation for Multi-objective Optimization. *Evolutionary Computation*, 15(1):1–28, March 2007.

[49] Erkki Oja. Oja learning rule. *Scholarpedia*, 3(3):3612, March 2008.

[50] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, January 2017.

[51] Stéphanie Ratté, Sungho Hong, Erik De Schutter, and Steven A. Prescott. Impact of Neuronal Properties on Network Coding: Roles of Spike Initiation Dynamics and Robust Synchrony Transfer. *Neuron*, 78(5):758–772, June 2013.

[52] Mark Steyvers, Michael D. Lee, and Eric-Jan Wagenmakers. A Bayesian analysis of human decision-making on bandit problems. *Journal of Mathematical Psychology*, 53(3):168–179, June 2009.

[53] Eric Schulz, Nicholas T. Franklin, and Samuel J. Gershman. Finding structure in multi-armed bandits. *Cognitive Psychology*, 119:101261, June 2020.

[54] Shunan Zhang and Angela J Yu. Forgetful Bayes and myopic planning: Human learning and decision-making in a bandit setting. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013.

[55] Jane X Wang. Meta-learning in natural and artificial intelligence. *Current Opinion in Behavioral Sciences*, 38:90–95, April 2021.

[56] Philippe N. Tobler, Christopher D. Fiorillo, and Wolfram Schultz. Adaptive Coding of Reward Value by Dopamine Neurons. *Science*, 307(5715):1642–1645, March 2005.

[57] Matthew R. Roesch, Donna J. Calu, and Geoffrey Schoenbaum. Dopamine neurons encode the better option in rats deciding between differently delayed or sized rewards. *Nature Neuroscience*, 10(12):1615–1624, December 2007.

[58] Roshan Cools. Chemistry of the Adaptive Mind: Lessons from Dopamine. *Neuron*, 104(1):113–131, October 2019.

[59] A. Aldo Faisal. Noise in Neurons and Other Constraints. In N. Le Novère, editor, *Computational Systems Neurobiology*, pages 227–257. Springer Netherlands, Dordrecht, 2012.

[60] João D. Nunes, Marcelo Carvalho, Diogo Carneiro, and Jaime S. Cardoso. Spiking Neural Networks: A Survey. *IEEE Access*, 10:60738–60764, 2022.

5 Appendix

5.1 Neural response function

The activation functions applied to the two neuronal population are defined as a step-function composed with a generalized sigmoid as follows:

$$f(x; g, o, \theta) = \begin{cases} \left[1 + e^{-g(x-o)}\right]^{-1} & \text{if } \left[1 + e^{-g(x-o)}\right]^{-1} > \theta \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

Where:

- x is the neuron pre-activation value
- g is the gain
- o is the offset
- θ is the threshold

Each population has its own set of parameters, which are optimized through evolutionary search.

5.2 Gaussian-sigmoid function

The function Φ. is defined by combining a generalized version of the sigmoid, namely with a gain $\beta \neq 1$ and offset $\alpha \neq 0$, and a Gaussian with mean μ and variance σ^2. Their contributions are weighted by as r and $1 - r$ ($r \in (0, 1)$) respectively.

$$\Phi_v(x) = r\left(1 + \exp^{-\beta(x-\alpha)}\right)^{-1} + (1-r)\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

The motivation behind this choice is to express a function that possesses a bounded region (depending on μ, σ) at a high/low peak (depeding on the value of γ_2), and a continuous transition to a constant value (depending on the steepness of the sigmoid β, shift α, and intensity γ_1).

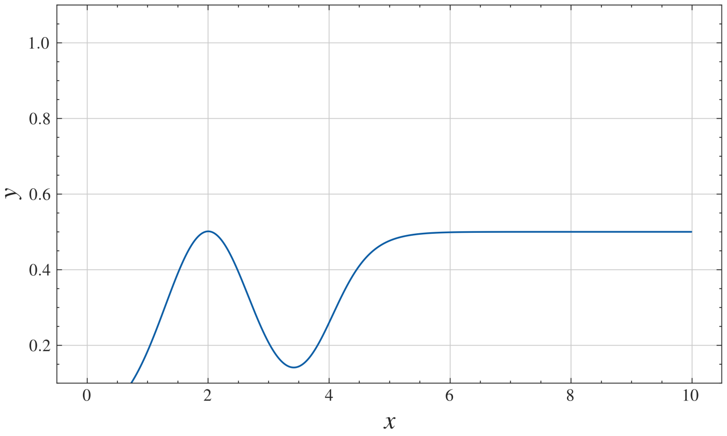

Figure 5: Activation function Φ_v - *Parameters* $\beta = 10$, $\alpha = 1$, $\mu = 1$, $\sigma = 1$, *and* $r = 0.5$.

5.3 Evolution search

The optimization was carried out over several parameters concerning the model architecture and dynamics:
Network parameters

- τ_u: time constant of population U
- τ_v: time constant of population V
- g_u: gain of the neural response function of population U
- g_v: gain of the neural response function of population V
- o_u: offset of the neural response function of population U
- o_v: offset of the neural response function of population V
- θ_u: threshold of the neural response function of population U
- θ_v: threshold of the neural response function of population V
- W^+: maximal weight value for the weights $\mathbf{W}^{UV}$

Option value function parameters

- β_v: steepness of the sigmoid
- α_v: shift of the sigmoid
- μ_v: mean of the Gaussian
- σ_v: variance of the Gaussian

- r_v: weight of the sigmoid

Learning rate function parameters

- β_η: steepness of the sigmoid
- α_η: shift of the sigmoid
- μ_η: mean of the Gaussian
- σ_η: variance of the Gaussian
- r_η: weight of the sigmoid

Each individual has been evaluated over environment the following environments:

- MAB-0: average reward distribution entropy $\langle H \rangle = 2.05$
- KAB-sinP: average reward distribution entropy $\langle H \rangle = 2.1$, given K arm frequencies f_k as an equally spaced set $\{0.1 \dots i \dots 0.4\}$, phases λ_k drawn from an uniform $\sim \mathcal{U}(0, 2\pi)$, and half of the arms have been set to constant values drawn from another uniform $\sim \mathcal{U}(0.1, 0.7)$; the final reward distribution was not normalized.

The number of arms was $K = 10$ and 150, and lasted for 2 trials with 2000 rounds each. The final fitenss was the average over 2 iterations.

The optimization has been implemented in Python using the `DEAP` library, and the algorithm used was the `CMA-ES` algorithm. The optimization involved 40 generations with a population size of 256 individuals. The mutation rate was set to 0.5 with a sigma of 0.8, the cross-over rate was set to 0.4. The run were carried out on a 256-core AMD EPYC 7763 with 2TB of RAM.

5.3.1 Genome distribution

Following the evolution search, it is taken the distribution of parameters over the top-scoring half of the population, corresponding to 128 individuals.

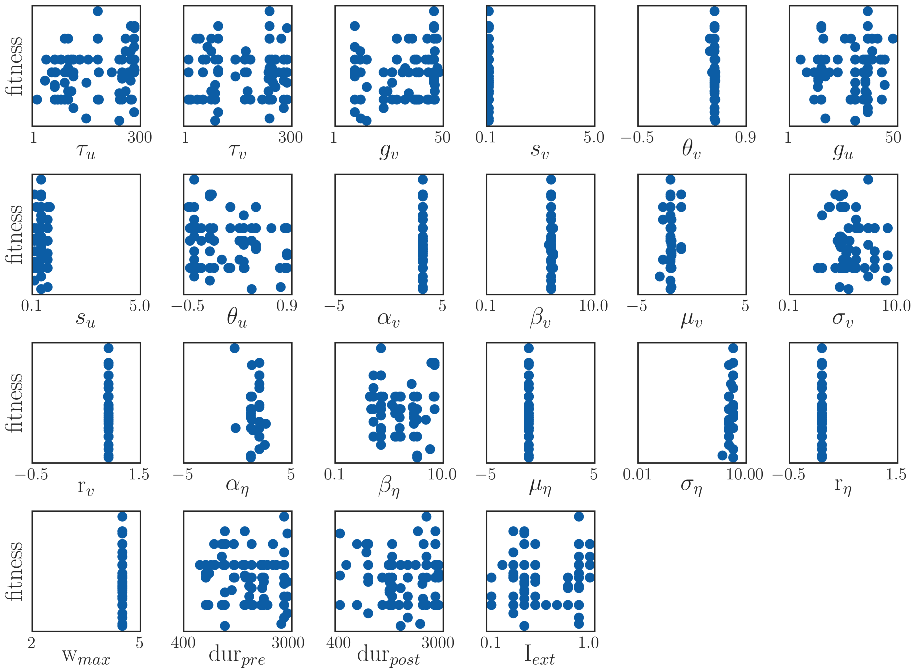

Figure 6: Genome distribution - *each parameter is plotted against the fitness score.*

Figure 6 reveals those parameters shared among the models with the highest fitness score, and those that are more variable. The most stable parameters are, predicatbly, those involved in directly shaping the neural activity (neural activation functions) and learning policy (Gaussian sigmoid).

5.4 Reward distribution entropy

The calculation of a set of N reward probability distribution $\mathbf{p}_i$ for $i \dots N$ for K values with a progressively decreasing levels of entropy $\mathbf{h}_i$ for $i \dots N$ has been obtained by the algorithm below 5.4.

Algorithm 2: Reward Probability Distribution Generation

```
Input: Number of distributions $N$, dimension $K$
Output: Set of probability distributions $\mathbf{p}_i$ with decreasing entropy
Initial Setup: Define set $B = \{17, 15, 12, 8, 4, 1.5, 0.5\}$;
for $i \leftarrow 1$ to $N$ do
    $\mathbf{z} \leftarrow \text{RandomVector} \sim \mathcal{U}(0, 0.5)^K$;
    $j \leftarrow \text{RandomIndex}(K)$;
    $\mathbf{z}_j \leftarrow 1$;
    $\beta_i \leftarrow$ Sample index=$i$ from $(B)$ ;  // Sample temperature from B
    $\mathbf{p}_i \leftarrow \frac{\exp(\beta_i \mathbf{z})}{\sum_j \exp(\beta_i \mathbf{z}_j)}$ ;  // Softmax with temperature
end
return $\mathbf{p}_i$
```

5.5 Weight update dynamics

We also analyzed the weight update dynamics of the model over the rounds. In figure 7, we plotted the evolution of the total weight ΔW^{UV} over time, averaged over 20 simulations and smoothed over 30 rounds. The results show that the model is able to quickly adapt to new reward distributions. It is also able to maintain the optimal policy over time, with the weights remaining approximately stable. The update quantity ΔW_k^{UV}, which at each round is applied to one connection k, changes sign according to the collected reward, with its magnitude being higher at the beginning of the trials. Initially, the sign is mostly positive (potentiation) since the weights start at zero, and after some uncertainty a consistently preferred arm emerges. However, when the reward distribution switches a regular series of sub-optimal choices with respect to the new distribution is made, leading to zero reward. This causes an accumulation of weight updates with negative sign (depression), eventually bringing the value of the preferred arm to drop. In the meantime, other options are probed until another sequence of choices converges to another arm, promoted by a trail of positive weight updates.

This behaviour is consistent with the low entropy levels observed in the previous analysis.

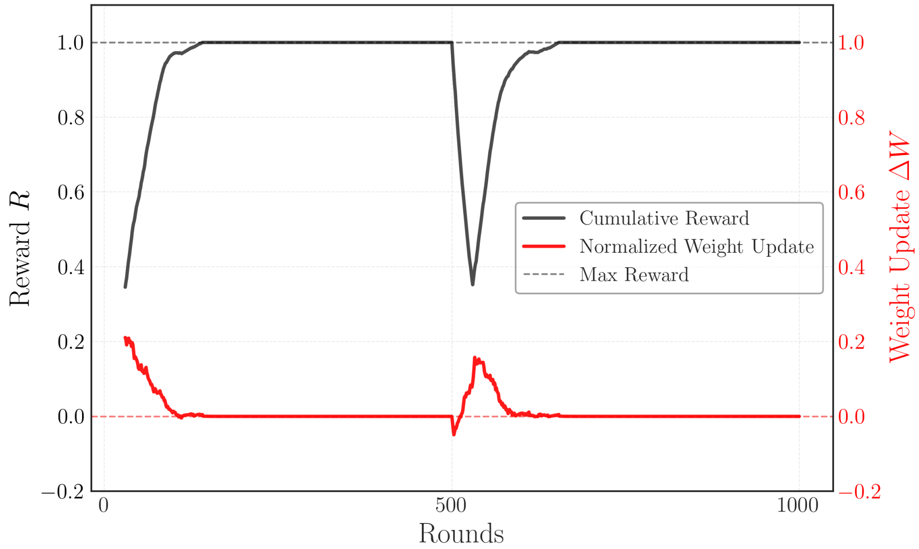

Figure 7: Weight update developement for the model *The plot displays the weight update quantity ΔW_k^{UV} for each round (blue line), smoothed as a 20-steps moving average. It is also reported the average reward in a window of 30 rounds (orange line). The results have been obtained averaging over 30 iterations.*