EasyChair Preprint

№ 14808

Deep Active Inference with Generative Actions and Diversity-Based Action Choice

Yacine Benabderrahmane, Godefroy Clair, Jérémy Dufourmantelle and Claire Ky

December 11, 2024

Deep Active Inference with Generative Actions and Diversity-based Action Choice

1st Yacine Benabderrahmane* 2nd Godefroy Clair 3rd Jérémy Dufourmantelle 4th Claire Ky
OCTO Technology *OCTO Technology* *OCTO Technology* *OCTO Technology*
Paris, France Paris, France Paris, France Paris, France
0009-0005-1933-4750 0009-0000-4565-1446 jeremy.dufourmantelle@octo.com claire.ky@octo.com
*Corresponding author

***Abstract*—The literature of Deep Active Inference, implementing the generative, biologically inspired Active Inference framework with the Deep Learning approach, often makes use of a hidden state transition model to generate current hidden states. It also usually leverages the Monte Carlo family methods to choose the agent's next action that minimizes the Expected Free Energy. The action identification typically uses either a stochastic sampling process or a learning of sampled actions by a 'habit' model. In this work, the goal is to explore an approach based on the learning and generation of actions as a result of hidden state transitions. The corresponding generative model, along with the variational form of the Free Energy and the Expected Free Energy, are formulated for an environment represented as a Partially Observable Markov Decision Process, and the model architecture is also presented. We also suggest a novel approach for the action choice: the generated action minimizing the Expected Free Energy is chosen based on the diversity of the expected risk relatively to that of its originating action set. The Active Inference agent is also equipped with top-down, selective, context-dependent attention mechanisms to control its behavior. Experiments have been conducted by addressing the continuous versions of Mountain Car and Inverted Pendulum problems. The results show the ability of the agent to learn and solve both problems with promising performance, requiring noticeable changes only on high-level attention parameters. This work highlights that this approach of action generation, choice and planning by Active Inference agents might represent a worthy alternative to usual methods, noticeably for considerations of computational efficiency and bio-mimetism.**

I. Introduction

Active Inference (AIF) is a generative Artificial Intelligence approach, initiated and developed as a conjunction of biomimetism and Information Theory [1]–[3], leveraging neuroscience findings on human and animal brain studies and built on the Free Energy (FE) minimization principle [4]. This framework is anticipated as a promising approach of generative AI [5] and it notably recognizes that the brain model implements generative processes [6] to achieve learning, reasoning and planing tasks. The relevance of this approach is also confirmed by experimental evidence, such as [7] which confirms that cellular and synaptic level behaviour are complying with the FE principle.

Deep Active Inference (DAIF) is one of the main approaches to implement AIF for solving problems at scale, initially addressing MDP (Markov Decision Process) environments [8]. Varying model architectures have been suggested, such as [9]–[11], validating also DAIF agent's ability to solve Partially Observable Markov Decision Processes (POMDP) [12], [13].

Presented briefly (see preceding references for more details), an AIF agent uses its perception model to make a sequence of observations $\tilde{o}$ of its environment's real states ($\tilde{x}$ being the sequence of a variable x through time). It uses these observations to build its own beliefs encoded in hidden states $\tilde{s}$ using its posterior model, by minimizing a homeostasis-ruled quantity, the FE, leveraging posterior reconstruction models. The agent interacts with the environment through actions $\tilde{a}$ executed by its actuator model, with the aim to enhance its beliefs about the world and to attain defined goals according to some provided priors. To choose the best action at a given time, it uses its internal beliefs to generate the likely consequences (observations $\hat{o}$) of actions it may take in the future, and select the action that minimizes the Expected Free Energy (EFE), a projection of the FE in the future.

The latter process is integrated in its action policy π as a planning of successive expected actions. Action planning efficiency may be one of the DAIF fields that needs deeper exploration as the implementations usually show a limitation in their capacity to plan. As each of the 3 main types of action selection policies (plan-based, habit and hybrid search, see [4]) brings advantages and drawbacks, different approaches have been suggested to achieve progress, *e.g.*: [9], [12] use bootstrapping of the EFE evaluation to enable its learning by a specific network, [14] use a "habit" policy to map states to actions and use it for minimizing the EFE, while others rely on Monte Carlo (MC) family methods to achieve the EFE minimization. To deal with the inherent computational

cost issue of MC approaches, MC Tree Search [15] has been used to optimize the search effort implied by the recursive reevaluation of the EFE at each time-step [10], [11], [16]. Some other authors rely on adapting the EFE formulation: while all of its components are used by [11], [17] truncate its formulation, [12] include in it the environment-provided reward and a discount value for future time-steps, and [10] add a hyper-parameter to allow a trade-off between reaching preferred states and resolving uncertainty.

There is still a debate on the relevant formulation of the EFE, which is suspected to lack in expected information gain [18]. Interestingly, a concept of action choice has been proposed by [19]. Given that the EFE is composed of risk and ambiguity parts (see section II-B), when an agent samples diversified actions to evaluate the corresponding EFEs, it will tend to choose the action that minimizes the risk to seek for prior realization if the risks are also diversified. Otherwise, the agent will tend to choose the action that will enable him a to improve its representation of the environment, *i.e.* the action that minimizes the ambiguity part.

Regarding the model architecture, DAIF agents often integrate posterior encoding models to implement approximations of the variational forms of the FE, and some use habit networks to learn an approximation of the EFE. Transition posterior models are typically used to learn the transition from past to present hidden states, such as in [13]. Besides, while using the continuous form of the model, discretizing the action space is often used to lower training complexity [11]–[13].

AIF agents can also be equipped with selective attention mechanisms, enabling the optimization of the agent's expected precision (or uncertainty) of states due to uncertainties and random fluctuations [20]. Learning rate modulation is viewed as a key feature underlying the selective attention mechanism [21]. [22] uses learning rate modulation linearly dependent on the prediction error magnitude, and many Bayesian approaches use it as a function of the variance (or uncertainty) in the current estimate of the predicted reward and in the reward values [23]. In the context of DAIF, [24] uses context-dependent precision parameters to generate top-down, task-dependent, selective attention. [11] implement a top-down attention mechanism by modeling the relation between the precision of state transitions, as described by [25], using a precision parameter based on the Kullback–Leibler divergence between learned and predicted actions by the action policy.

In this work, we explore the ability of a DAIF agent, equipped with top-down attention mechanisms, to learn and generate its actions and to solve typical continuous AI POMDP environments, using a transition model to learn actions from state transitions in a continuous action space. Due to the inferred nature of the actions, a habit-like action policy is used, leveraging an action choice strategy based on the diversity of generated actions and their expected consequences. We use an EFE formulation based only on observed quantities, free of empirical weighting hyper-parameters or environment reward. The overall goal of this paper is to experiment these new approaches and analyze the results, rather than perform a benchmark against other approaches which can perform far better at this stage.

II. Active Inference and Generative Actions

A. Generative Model

To integrate the concept of generative actions as resulting from hidden state transition, and considering the environment as a POMDP, the generative model $P(\tilde{o}, \tilde{a}, \tilde{s}, \pi)$ built by the agent is a joint probability formulated as:

$$P(\tilde{o}, \tilde{a}, \tilde{s}, \pi) = P(\pi)P(s_0)\prod_{t=1}^{T} P(o_t \mid s_t) \times P(s_t \mid s_{t-1}, a_{t-1})P(a_{t-1} \mid s_t, s_{t-1}, \pi) \quad (1)$$

where $P(\pi)$ are the beliefs about policies, $P(o_t \mid s_t)$ is the likelihood mapping and $P(s_t \mid s_{t-1}, a_{t-1})$ and $P(a_{t-1} \mid s_t, s_{t-1}, \pi)$ are posterior transition probabilities of hidden states and generated actions, respectively. Notably, the model integrates here the agent actions as part of the sensory information, and the transition probabilities of actions are conditioned on the current and preceding generated hidden (latent) states. This provides the agent the ability to build beliefs about posterior actions based on the transition from the previous to the current hidden states.

B. Variational formulation

The minimization of the agent's surprise in AIF relies on the minimization of the FE which is an upper bound of the former quantity. The variational free energy is an approximation of the FE and has many derivations [3], [4]. We choose here, in a discrete-time context for continuous spatial and action spaces, the complexity / accuracy decomposition:

$$F = D_{KL}[q(s_t) \| \, p(s_t)] - \mathbb{E}_{q(s_t)}[\log p(o_t \mid s_t) + \log p(a_{t-1} \mid s_t, s_{t-1})] \quad (2)$$

where $q(s_t)$ is an approximation of the variational posterior over states, $p(s_t)$ is the prior model over hidden states, and $p(o_t \mid s_{t-1})$ and $p(a_{t-1} \mid s_t, s_{t-1})$ are the likelihood models generating respectively, given the hidden states, distributions of sensory observations and actions. Crucially, we call the latter the 'transition model' as it reconstructs action distributions while integrating their consequences on the transition between

the preceding and the current states, s_{t-1} and s_t. The Kullback–Leibler divergence term (D_{KL}) represents an approximation of the generative model complexity which forces the posterior model to generate a distribution of hidden states as close as possible to that of the model prior. The second term is the approximation of the generative model accuracy, maximizing the expectation of sensory observations and actions reconstruction over hidden states. This makes minimizing (2) equivalent to finding the trade-off between model beliefs and reconstruction in Variational Autoencoders [26].

To take an action in the future that enhances its knowledge about the environment and/or promotes the realization of the preferred perceptions (observations and actions), the agent must minimize the surprise in the future (therefore the free energy in the future), using its generative model to imagine the subsequent observations of the actions it may generate. To achieve this, the agent applies action policies corresponding to its beliefs, each policy π resulting in a value of the expected free energy $G(\pi)$ at a given time horizon T, and chooses to perform the action that is expected to produce the lower G:

$$\begin{aligned} P(\pi) &= \sigma(-G(\pi)) \\ G(\pi) &= \sum_{\tau=t}^{t+T} G(\pi, \tau) \end{aligned} \tag{3}$$

where $P(\pi)$ is the probability distribution of a policy π and σ is the softmax function. The action policies used in our context are detailed in section III.

The formulation of the expected free energy in our context is, for a future time-step τ, and using the expected ambiguity / risk derivation:

$$\begin{aligned} G(\pi, \tau) = & D_{KL}[q(o_\tau \mid \pi) \| \, p(o_\tau)] \\ & + D_{KL}[q(a_{\tau-1} \mid \pi) \| \, p(a_{\tau-1})] \\ & + \mathbb{E}_{q(s_\tau \mid \pi)}[H[p(o_\tau \mid s_\tau)] \\ & + H[p(a_{\tau-1} \mid s_\tau, s_{\tau-1})] \,] \end{aligned} \tag{4}$$

where $p(o_\tau)$ and $p(a_{\tau-1})$ are respectively the agent prior belief on the observation and the action distributions. The first two terms of this equation form the risk R over the outcomes (expressed as the D_{KL} between the approximated posteriors and the prior beliefs), while the last term refer to the expected ambiguity A (formulated as expectations $\mathbb{E}$ of outcome and action entropies H).

C. DAIF Model Architecture

The DAIF model architecture implementing our model (see Fig. 1) is mainly based on amortization and reparameterization techniques. Neural network models are used to approximate the variational models: $q_\phi(s_t \mid s_{t-1}, o_t, a_{t-1})$ acts as a hidden state encoder for the posterior model $q(s_t \mid s_{t-1}, o_t, a_{t-1})$, while $p_\xi(o_t \mid s_t)$ and $p_\theta(a_{t-1} \mid s_t, s_{t-1})$ acts as outcome decoders for the observation likelihood model $p(o_t \mid s_t)$ and the transition model $p(a_{t-1} \mid s_t, s_{t-1})$, respectively. Here, ϕ, ξ and θ are respectively amortization parameters for q_ϕ, and network parameters for p_ξ and p_θ.

Training these networks simultaneously in continuous learning mode boils down to minimize for each time-step t the objective function corresponding to the variational FE (2):

$$\begin{aligned} F_t = & D_{KL}[q_\phi(s_t) \| \, p(s_t)] \\ & - \mathbb{E}_{q_\phi(s_t)}[\log p_\xi(o_t \mid s_t) \\ & + \log p_\theta(a_{t-1} \mid s_t, s_{t-1})] \end{aligned} \tag{5}$$

where expectations over q_ϕ are computed using a single sample from the encoder. The EFE defined in (3) and (4) is computed at each time-step using these models' inference in the future (see also section III) with:

$$G_t(\pi) = R_t(\pi) + A_t(\pi) \tag{6}$$

the risk R_t and ambiguity A_t parts being:

$$\begin{aligned} R_t(\pi) = & \sum_{\tau=t}^{t+T} [D_{KL}[q_\phi(o_\tau \mid \pi) \| \, p(o_\tau)] \\ & + D_{KL}[q_\phi(a_{\tau-1} \mid \pi) \| \, p(a_{\tau-1})] \,] \\ A_t(\pi) = & \sum_{\tau=t}^{t+T} \mathbb{E}_{q_\phi(s_\tau \mid \pi)}[H[p_\xi(o_\tau \mid s_\tau)] \\ & + H[p_\theta(a_{\tau-1} \mid s_\tau, s_{\tau-1})]] \end{aligned} \tag{7}$$

The implementation assumes that s_t, o_t and a_t are normally distributed, and that each model outputs multivariate normal distributions with diagonal covariance matrices, making the D_{KL}, log-likelihood and entropy terms easy to compute with standard expressions.

The sensory model alters the agent's sensory input in two ways. It first integrates perturbations to sensory observations by random noise addition $\varepsilon_o \sim \mathcal{N}(0, 1)$ to each observation o^* using $o = o^* + \varepsilon_o$, while the agent is supposed to have a perfect perception of the actual actions applied on the environment by its actuator. It then normalizes the observations using the physical domain characteristics to feed the generative model with observations centered on 0 in the range $[-1, 1]$ and actions in the range $[-1, 1]$. This last operation ensures the model ability to address different environments and a better fit to deep neural networks' operational domain.

Actual action a_t^*, time-rolling to a_{t-1}^*, is applied by the actuation model on the environment, after unnormalizing the chosen action $\hat{a}_t$ and applying its own physical constraints and those of the environment.

This architecture enables the transition model to generate the distribution of actions $\hat{a}_{t-1}$ that ensure the likelihood maximization of the preceding action a_{t-1} using (5), while accounting for the transition between the preceding hidden states s_{t-1} and the current generated

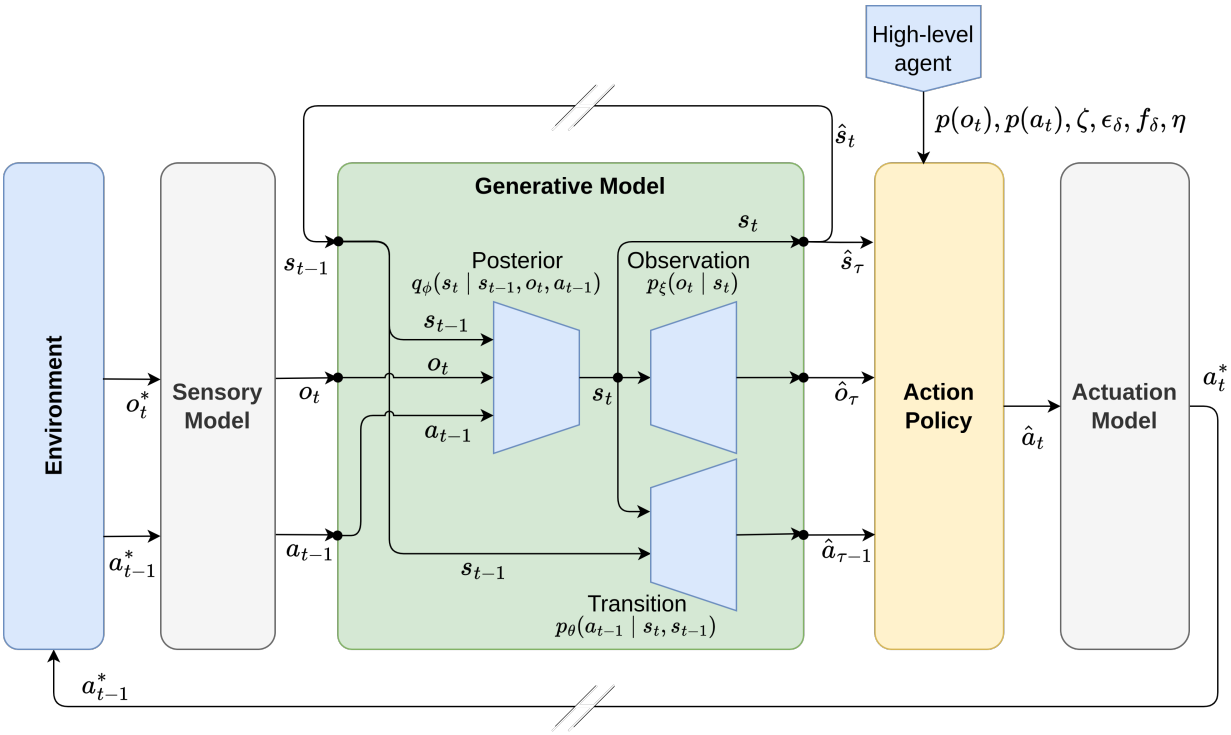

Fig. 1: Agent's model components overview, where all neural networks of the generative posterior $q_\phi(s_t \mid s_{t-1}, o_t, a_{t-1})$, observation $p_\xi(o_t \mid s_t)$ and transition $p_\theta(a_{t-1} \mid s_t, s_{t-1})$ models are trained simultaneously. The transition model learns actions as hidden state transitions.

hidden states s_t. The latter mapping also makes the agent learn the actuator behavior and operational domain (*e.g.* out-of-domain generated actions result in no change in perceived observations and actions, and in corresponding hidden states).

The overall generative model, when invoked by the action policy, acts as an inference machine to generate, based on its learned beliefs, expected states s_τ and expected outcomes $\hat{o}_\tau$ and $\hat{a}_\tau$ for a given time-step τ. The high-level agent provides prior preferences for observations $p(o_t)$ and actions $p(a_t)$, as well as functional hyper-parameters ζ, ϵ_δ, f_δ and η (discussed below).

III. Diversity-based Action Choice

The model described in section II implies an action selection policy where actions are not chosen at each time-step, but are instead generated as a direct consequence of the expected hidden states and expected observations (see Fig. 2). This can be viewed as an extension of 'habit' action policies [4], with multiple simultaneous policies.

When a set $\alpha_t = \{\hat{a}_{t,i}\}$ of N actions is generated at time-step t ($i \in [1, N]$), the sets of risk values $\mathcal{R}_t = \{R_{t,i}\}$ and ambiguity values $\mathcal{A}_t = \{A_{t,i}\}$ are computed using (7). Following the idea of [19], we define an action choice strategy based on the comparison between the diversities (defined below) of these sets. If the generated action set α_t and the risk set $\mathcal{R}_t$ have comparable diversities, the agent is believed to be more comfortable in choosing the action $\hat{a}_{t,i}$ resulting in a risk $R_{t,i}$ which minimizes the risk set $\mathcal{R}_t$. Otherwise, the agent will choose the action that minimizes the ambiguity set $\mathcal{A}_t$. When choosing the action to minimize the risk set, the agent decides that it has enough good representation of the world and seeks for realizing extrinsic value. When it chooses to minimize the ambiguity, it wants to improve its hidden representation of the world (intrinsic information value).

Diversity is defined as a measure of similarity between the values of a set. Among the most commonly used diversity metrics, the diversity of [27] is category-oriented, and its adaptation to real-valued sets is not straightforward. The Vendi score [28] provides good sensitivity to the dissimilarity of real values, but has the drawback of being an unbounded estimator, making it of difficult use in our context. We formulate here a bounded diversity estimator, $\delta \in [0, 1]$, suitable for reasonably small, real-valued sets. Its maximum value $\delta_{\max} = 1$ corresponds to a set whose values are as much as possible different from one another, and its minimum value 0 corresponds to the least diversified sample, *i.e.* all elements having the same constant value. For a one-dimensional sample x of size $N > 1$, the diversity estimator $\delta(x)$ is defined

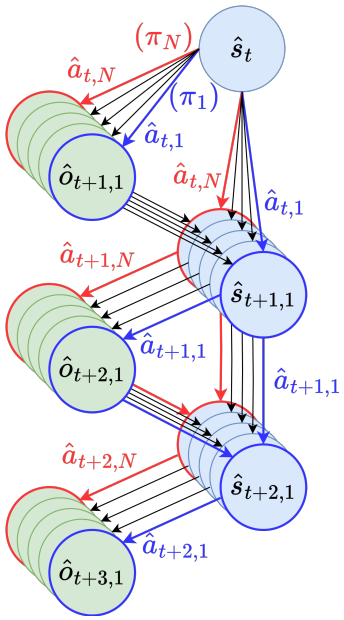

Fig. 2: Action policy applied by the agent. From an initial hidden state sample $\hat{s}_t$ at time-step t, a set of N actions $\{\hat{a}_{t,i=1..N}\}$ is generated. Using this set and $\hat{s}_t$, the generative model infers simultaneously the corresponding sets of likely observations $\{\hat{o}_{t+1,i=1..N}\}$ and hidden states $\{\hat{s}_{t+1,i=1..N}\}$, which are in turn used to generate the next time-step's actions $\{\hat{a}_{t+1,i=1..N}\}$. The process is repeated until reaching the desired time horizon T ($T = 2$ here). Each sequence $\{\hat{a}_{\tau,i}\}$, $\tau \in [t, t+T]$ is a separate action policy π_i, executed with no intermediary action choice.

as:

$$\begin{aligned}\delta(x) &= \frac{\sigma(x)}{\nu(x)}\,,\ \nu(x) \\ &= \left(\frac{N+1}{12}\sum_{i=1}^{N-1}\left(x_{i+1}-x_i\right)^2\right)^{1/2}\end{aligned} \tag{8}$$

where $\mu(x)$ and $\sigma(x)$ are the mean and the standard deviation of x, respectively. The maximum diversity $\delta_{max} = \delta(x^*) = 1$ is obtained for a set x^* composed of equally spaced values $\in [x^*_{\min}, x^*_{\max}]$ when sorted in increasing direction ($x^*_i = x^*_{\min} + \Delta\,(i-1)\,, i \in [2, N], \Delta > 0$). By convention, for a set x_c of identical values and $\forall N$, the minimum diversity $\delta_{\min} = \delta(x_c) = 0$.

The diversity ratio $\bar{\delta}$ between the risk set and the generated action set diversities, respectively $\delta_{\mathcal{R}} = \delta(\mathcal{R}_t)$ and $\delta_\alpha = \delta(\alpha_t)$, is used to compare the diversities and is defined as:

$$\bar{\delta} = \delta_{\mathcal{R}}/\delta_\alpha \tag{9}$$

To identify the range inside which a specific quantity is to be minimized with respect to the diversity ratio, we define diversity ratio bounds:

$$\epsilon_{\delta,\min} \in [0,1],\ \epsilon_{\delta,\max} \in [0,+\infty[\tag{10}$$

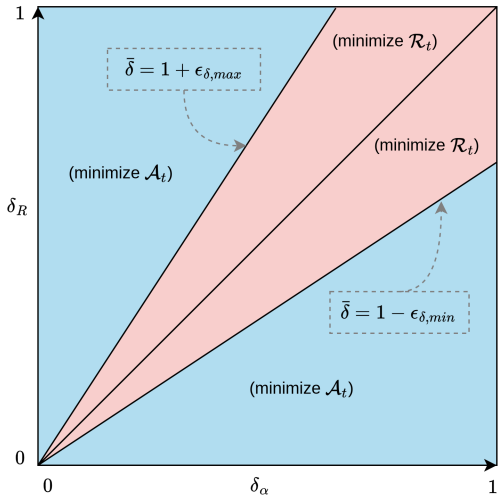

Fig. 3: Decision surface for the diversity ratio based strategy. If the diversity ratio $\bar{\delta} \in [1-\epsilon_{\delta,min}, 1+\epsilon_{\delta,max}]$ (red zone), the agent chooses the action minimizing $\mathcal{R}_t$, otherwise it chooses the one minimizing $\mathcal{A}_t$ (blue zone).

The Fig. 3 shows the decision surface depicting the agent's strategy: as the agent prefers risk and action set diversities that are near, and tolerates risk sets more diversified than action sets, the agent chooses the action which minimizes the risk set $\mathcal{R}_t$ if the diversity ratio $\bar{\delta}$ is near or over 1, with a tolerance $1+\epsilon_{\delta,\max}$. Similarly, the agent tolerates risk diversities slightly below 1 by $\epsilon_{\delta,\min}$. Otherwise, the agent chooses the action which minimizes the ambiguity set $\mathcal{A}_t$. These bounds are hyper-parameters that could be viewed as a qualification of the agent's behaviour in its action choice: the higher are $\epsilon_{\delta,\min}$ or $\epsilon_{\delta,\max}$, the more the agent is 'ambitious', having a higher probability of choosing actions that minimize the risk.

IV. Top-down Attention

To integrate top-down selective attention, we introduce precision parameters ζ_o and ζ_a in the agent's generative model, following [25], to equip the agent with beliefs about the uncertainty in likelihood mapping from hidden states to observations and actions, respectively. The Free Energy equation (2) becomes:

$$\begin{aligned}F = D_{KL}[q(s)\|\,p(s)] - \mathbb{E}_{q(s)}[\zeta_o\,\log p(o \mid s) \\ + \zeta_a\,\log p(a_{t-1} \mid s, s_{t-1})]\end{aligned} \tag{11}$$

where the current time t subscript is omitted for readability. Similarly, integrating expected precision over

outcomes using the precision parameters $\zeta_{\mathrm{e},o}$ and $\zeta_{\mathrm{e},a}$ gives the new form of (4):

$$\begin{aligned} G(\pi,\tau) = \zeta_{\mathrm{e},o}\, D_{KL}[q(o_\tau \mid \pi) \| \, p(o_\tau)] \\ + \zeta_{\mathrm{e},a}\, D_{KL}\left(q\left(a_{\tau-1} \mid \pi\right) \| \, p\left(a_{\tau-1}\right)\right) \\ + \mathbb{E}_{q(s_\tau|\pi)}[\zeta_{\mathrm{e},o}\, H\left[p(o_\tau \mid s_\tau)\right] \\ + \zeta_{\mathrm{e},a}\, H\left[p(a_{\tau-1} \mid s_\tau)\right]\,] \end{aligned} \tag{12}$$

Crucially in our work, these parameters are state-specific, each state having its own precision and expected precision parameter.

We also equip the agent with context-dependent, top-down attention through attention parameters provided by the high-level agent. In addition to the definition of preferred prior distributions, to guide the active inference agent into attaining the assigned goal of the task, the high-level agent evaluates the agent's performance given the goal, defines the learning rate of the agent with respect of its performance on the task and defines steering the agent's action policy to attain the goal.

The agent's performance on a task in an environment E is modeled as a score function $\mathcal{S}$ of the observations, the preferred prior distributions and a high-level, task dependent, objective $\mathcal{O}(E,t)$, in a time window w:

$$\begin{aligned} \mathcal{S} = \mathcal{S}\left(o_\tau, p(o_\tau), p(a_\tau), \mathcal{O}(E,\tau)\right), \\ \tau \in [t-w, t]\,,\ \mathcal{S} \in [0,1] \end{aligned} \tag{13}$$

If the task involves reaching a goal in the context of episodes, the performance is often impacted by a high-level evaluation of each episode's observations (*e.g.* number of steps before reaching the goal or number of steps maintaining the observations in the goal range).

The agent is equipped with top-down learning rate modulation expressed using an exponential decay from a initial learning rate η_{init} as a function of the performance score:

$$\eta = \eta_{\mathrm{init}} \exp(-\lambda_\eta\, \mathcal{S}) \tag{14}$$

This way, the nearest agent is to the goal, the lower is its learning rate, the high-level agent considering that the performance is good enough and that it prefers avoiding jeopardizing the learning performance such as agent over-fitting or catastrophic interference [29]. This modulation can be viewed as an implementation of context-dependent (task-dependent), top-down selective attention applied to the update of the hidden states belief, driven by prediction errors as hinted by [24] and [23].

Similarly, the nearest the agent is to the goal, the more it will tend to choose actions that favor the expected extrinsic value over the expected intrinsic value, *i.e.* it considers the ambiguity minimization as less important relatively to that of the risk, making it more 'ambitious'. Its action strategy becomes more tolerant towards minimizing the expected risk over ambiguity, and practically, the agent tolerates more a diversity ratio farther from 1. This is modeled by adjusting the diversity bounds (10):

$$\begin{aligned} \epsilon^{'}_{\delta,\min} &= \epsilon_{\delta,\min} + f_\delta\left(\epsilon_{\delta,\min,\mathrm{adj}} - \epsilon_{\delta,\min}\right) \\ \epsilon^{'}_{\delta,\max} &= \epsilon_{\delta,\max} + f_\delta\left(\epsilon_{\delta,\max,\mathrm{adj}} - \epsilon_{\delta,\max}\right) \end{aligned} \tag{15}$$

where $\epsilon_{\delta,\min,\mathrm{adj}}$ and $\epsilon_{\delta,\max,\mathrm{adj}}$ are hyper-parameters specifying the extreme values the adjusted ϵ_δ defined in (10) can take, and f_δ is the diversity adjustment factor, itself a monotonically increasing function of the performance score:

$$f_\delta = f_\delta(\mathcal{S})\,,\ f_\delta \in [0,1] \tag{16}$$

V. Experiments & Results

Experiments consisted in solving the Open AI Gym Mountain Car Continuous v0 [30] and Mujoco Inverted Pendulum v4 [31] environments, with their default configurations. Both environments truncate the episodes when 10^3 steps are reached. The agent was asked to solve these continuous problems in a continuous learning mode, making the agent interacting with the environment, learning and inferring at each time-step. Only the environment states were reset at the beginning of each episode, while a completely new agent was used for each new experiment.

Prior preference over observation distributions were fixed, over all observations and time, to normal distributions: $p(o_i) \sim \mathcal{N}(\mu_{p,o,i}\,, \sigma_{p,o,i})$. Whilst the action prior preference $p(a_{\tau-1})$ in (12) was discarded using the top-down attention parameter $\zeta_{e,a} = [0]$ for both environments, allowing the agent to freely generate actions but still learning actuator domain and state transitions, the selection of expected observations through the parameters $\zeta_{e,o}$ was problem-specific. Therefore, defining a prior distribution was only needed for selected observations. For both problems, the high-level agent relied on the end of episode evaluation to steer the DAIF agent behavior at the higher level (via the top-down attention mechanism), *i.e.* did not integrate the reward at each step in the EFE, but evaluated the performance as the achievement of the entire experience in the entire episode.

The same model architecture was used for both problems, using 2-layer neural networks of size 20 for each of the posterior, observation likelihood and transition models. The latent dimensions of encoded hidden states were problem dependent. The Adam optimization algorithm was used to minimize (5) as the main loss with respect to the model parameters ϕ, ξ and θ, completing one epoch per time-step.

Three baseline agents were used to enable model evaluation: *i)* a 'random action agent' was asked to solve the problem by performing an action randomly sampled from a uniform distribution ($\hat{a}_t \sim \mathcal{U}(-1,1)$), *i.e.* not

using its transition model to infer actions; *ii)* a dummy 'no action agent' was not allowed to perform actions ($\hat{a}_t = 0$); *iii)* a 'random generated action agent' choosing randomly the action from the generated action set α_t, instead of applying the diversity-based action choice strategy described in section III.

A. Mountain Car

The Mountain Car is a typical complex problem used in AIF and reinforcement learning to assess agent ability to learn the environment dynamics. The goal for the agent is to move a car to a goal at the top of a mountain (see Fig. 4). This goal can only be reached if the car is first moved to the opposite side of the mountain to gain momentum, forcing the agent to build an internal model integrating the physical laws of the environment to succeed.

The environment provides a bi-dimensional car state vector [position, velocity]. Both were selected as observed states in (11), along with the action observation ($\zeta_o = [1, 1]$, $\zeta_a = [1]$). The expected position and velocity were also selected in expected observations of (12) ($\zeta_{e,o} = [1, 1]$). The prior preferences on states were fixed to $p(o_1) \sim \mathcal{N}(\mu_{p,o,1} = 1\,, \sigma_{p,o,1} = 0.1)$ and $p(o_2) \sim \mathcal{N}(\mu_{p,o,2} = 0.08\,, \sigma_{p,o,2} = 3)$. The latent dimensions of encoded hidden states were set to [4, 2]. The experiments were run with a maximum duration of $24\,10^3$ episodes.

The task-dependent, high-level objective function set for this problem is a success condition at the granularity of the episodes, where the agent must achieve $n_{\text{sse}} = 10$ successive success episodes n_{se}:

$$\mathcal{O}(o,t) = \mathcal{O}(n_{\text{se}}, n_{\text{sse}}) = \mathcal{H}\left(\sum_{k=1}^{n_{\text{sse}}} n_{\text{se},k} - n_{\text{sse}}\right) \quad (17)$$

where $\mathcal{H}$ is the Heaviside function, while the score function (13) was simply set to $\mathcal{S}(\mathcal{O}) = 0$, the diversity adjustment factor (16) was set to $f_\delta = 0$ and the empirically identified values in (14) and (15) were: $\eta_{\text{init}} = 1.5\,10^{-4}$, $\lambda_\eta = 0$ (no decay), $\epsilon^*_{\delta,min} = 0.4$ and $\epsilon^*_{\delta,max} = 4$. Whenever the success condition was met (*i.e.* $\mathcal{O} = 1$), the agent was allowed to stop learning and to switch to inference only mode.

The results of simulations, executed with a time horizon $T = 2$ and $N = 10$ action policies, are shown in Fig. 4. The 'random action agent' experience resulted in over 800 steps per episode of in average for 1000 episodes (in red in Fig.). The 'random generated action agent' succeeded in solving the problem with around 250 steps per success episode and a large standard deviation of about 60 (in yellow in the Fig.). When using the diversity-based action choice strategy, and although requiring a longer learning time than the latter experiment, the agent was able to solve the problem with notably better performance (between 100 and 210 steps per episode) and higher consistency through episodes, the best experience (in blue) showing even a negligible standard deviation.

The Fig. 5 shows the phase diagrams corresponding to a complete successful episode of the best performing agent (blue in Fig. 4). A typical sequence of forth, back and forth of the car can be seen in the phase state diagram, with a sharp deceleration at the extreme left position. The corresponding actions exhibit a slightly noisy evolution although remaining consistent. They attain the maximum value for many time-steps when the agent reaches out for momentum on the mountain and when it heads for the goal.

B. Inverted Pendulum

The Inverted Pendulum Problem is a classical, complex control task where a pole is positioned on a cart which slides along a horizontal axis (see Fig. 6). The goal is to keep the pole upright within a tolerated angle range specified by the environment (± 0.2 radians in our case), otherwise the environment truncates the episode. A key difference between this problem and the preceding one is that the episodes start with the pole very near to the goal, and the challenge for the agent in a continuous learning mode is to learn the environment dynamics from much shorter episodes during many episodes in the beginning of the experiment, and the longer the episode, the more it is exposed to new states to learn. This results in a more complex problem to solve for the agent.

The environment provides a four-dimensional state vector [pole angle, cart position, pole angle velocity, cart velocity]. All but the third state were selected as part of the observed states in (11), along with the action observation ($\zeta_o = [1, 1, 0, 1]$, $\zeta_a = [1]$), and only the pole angle was selected for the expected observations in (12) ($\zeta_{e,o} = [0, 1, 0, 0]$). The prior preferences over these states were fixed to $p(o_2) \sim \mathcal{N}(\mu_{p,o,2} = 0\,, \sigma_{p,o,2} = 0.025)$. The latent dimensions of encoded hidden states were set to [6, 2]. The experiments were run with a maximum duration of $4\,10^4$ episodes and $35\,10^4$ steps.

The task-dependent, high-level objective function set for this problem is a success condition at the episode level, where the agent must achieve $n_{\text{sse}} = 5$ successive episodes of more than $n_{\text{es}} = 35$ steps:

$$\begin{aligned}\mathcal{O}(o,t) &= \mathcal{O}(n_{\text{spe}}, n_{\text{sse}}, n_{\text{es}}) \\ &= \mathcal{H}\left(\sum_{k=1}^{n_{\text{sse}}} n_{\text{spe},k} - n_{\text{es}}\right)\end{aligned} \quad (18)$$

The score function (13) is modeled as a simple moving average of the number of steps per episode n_{spe} over n_{sse} successive success episodes, conditioned by the objective

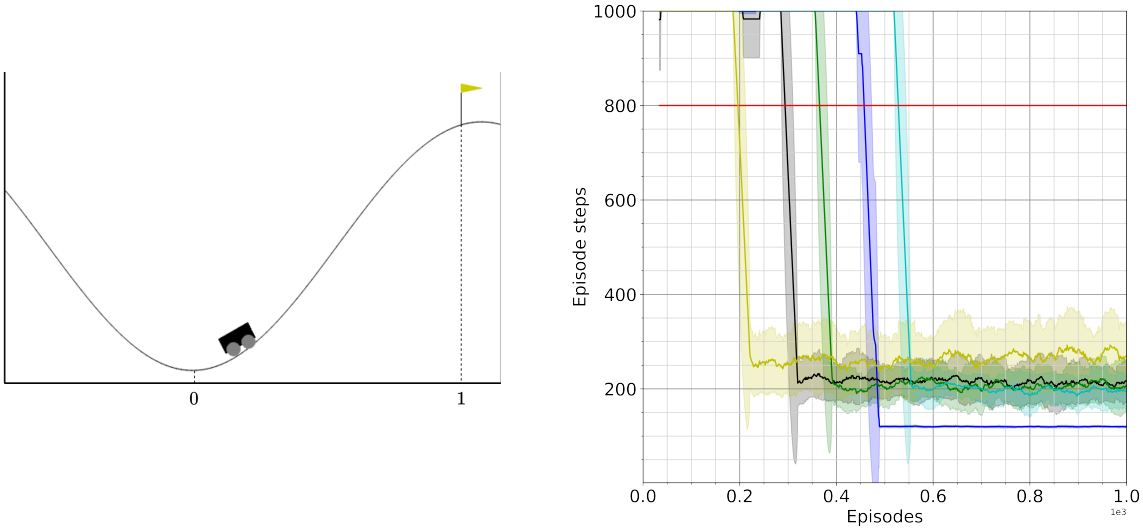

Fig. 4: Left: Mountain Car environment (position is normalized). Right: Average episode steps per episode (moving average over 35 steps, bold lines) along with the corresponding standard deviations (filled areas) of agents solving the Mountain Car problem (truncated to 10^3 first episodes). The 'random action agent' (averaged, in red) and 'random generated action agent' (yellow) experiences shown for reference. The diversity-based agents (black, green, blue, cyan) outperform baseline agents and are more consistent compared to the 'random generated action agent', although requiring longer learning. They can even achieve noticeable performance and consistency, as in (blue).

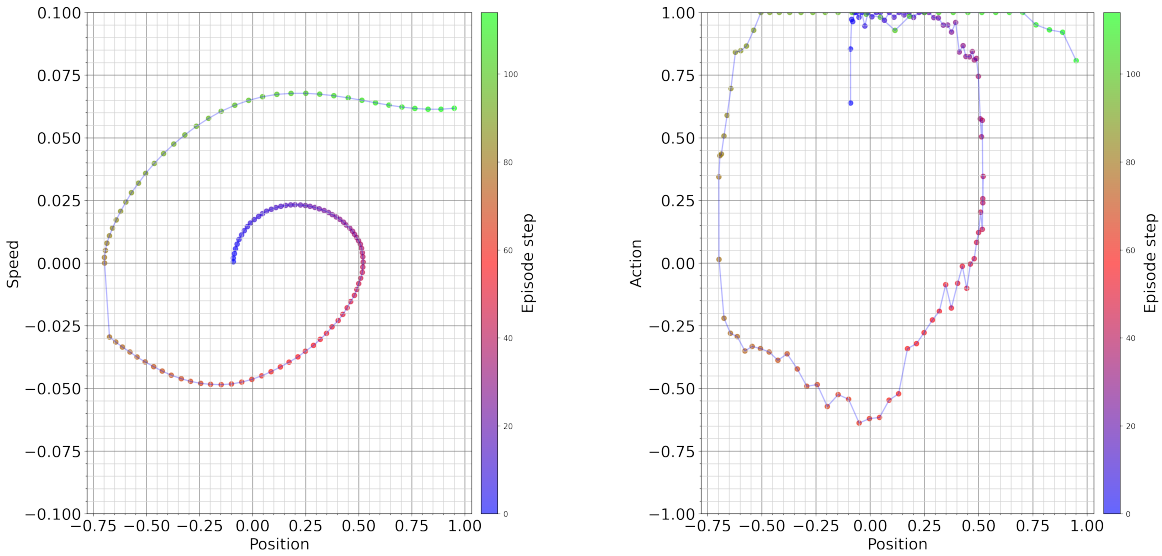

Fig. 5: Complete episode phase diagrams of the agent solving Mountain Car problem (from best performing agent, in blue in fig. 4). Left: phase state diagram, right: corresponding action vs position phase diagram (position is normalized).

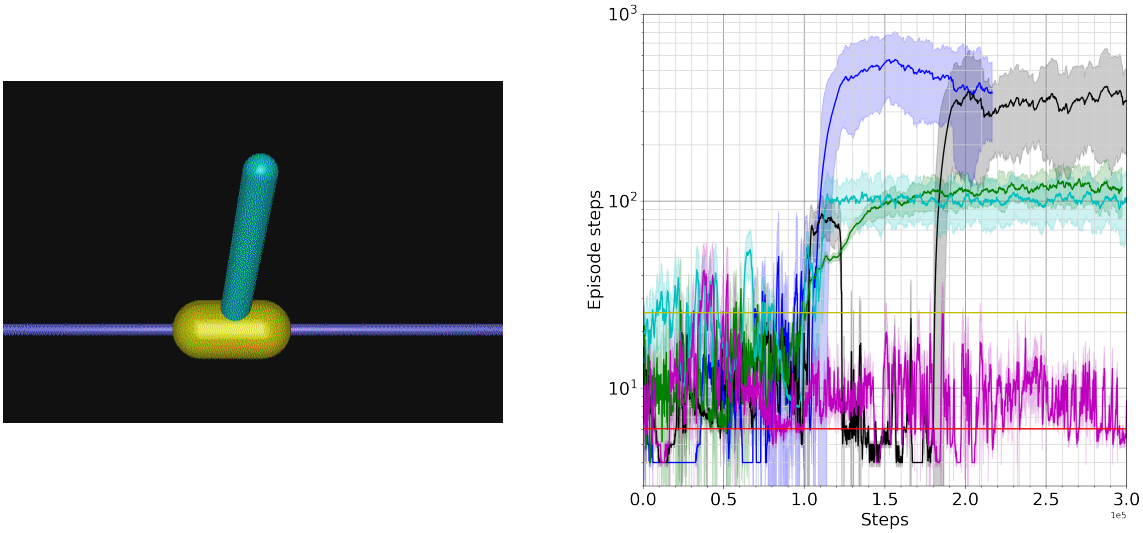

Fig. 6: Left: Inverted Pendulum environment. Right: Average episode steps vs steps (moving average over 35 steps, bold lines) along with the corresponding standard deviations (filled areas) of agents solving Inverted Pendulum problem. The experiences of the 'random action agent' (averaged, in red) and 'no action agent' (averaged, in yellow) are shown for reference. The agent 'random generated action agent' (magenta) failed systematically in solving the problem. Diversity-based agents succeed with various performances (blue, black, green and cyan). The agent in (black) shows the ability to recover stable problem solving performance after a first phase of success followed by a failure phase.

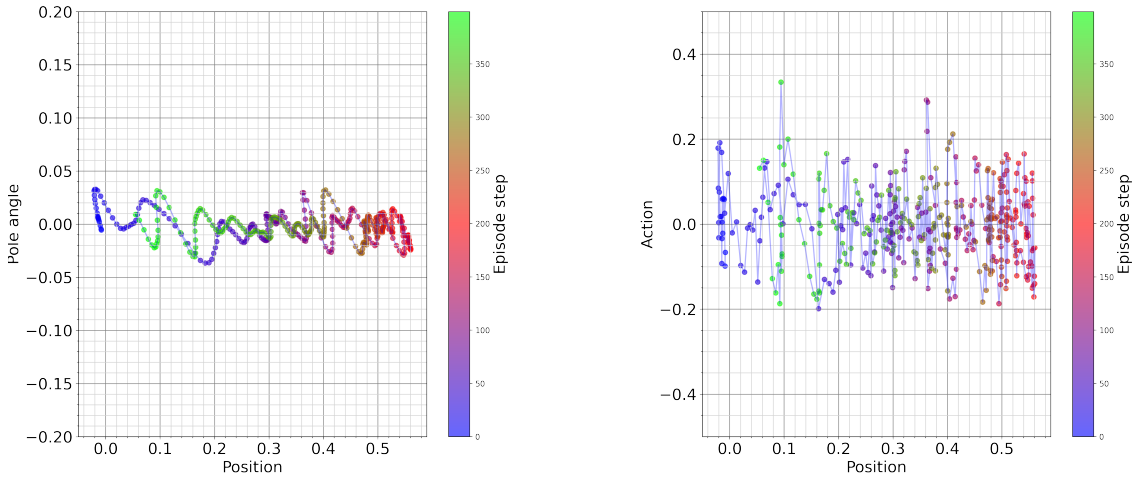

Fig. 7: Phase diagrams of the agent solving Inverted Pendulum problem (showing first 400 steps of a successful episode from best performing agent in blue in fig. 6). Left: phase state diagram, right: corresponding action to position phase diagram.

function and weighted by the environment truncation $n_{\text{env,max}} = 1000$:

$$\begin{aligned}\mathcal{S}(\mathcal{O}) &= \mathcal{S}(n_{\text{spe}}, n_{\text{sse}}, n_{\text{es}}) \\ &= \mathcal{O}(n_{\text{spe}}, n_{\text{sse}}, n_{\text{es}}) \, \frac{\text{SMA}_{n_{\text{sse}}}(n_{\text{spe}})}{n_{\text{env,max}}}\end{aligned} \tag{19}$$

where $\text{SMA}_k(n_{\text{spe}}) = \frac{1}{k} \sum_{i=n-k+1}^{n} n_{\text{spe},i}$.

Whenever the learning rate η reached the minimum value $\eta_{\min}$ (10^{-6} in our experiments), the agent was allowed to stop learning and to switch to inference only mode. Switching back to learning was allowed if ever the success condition was not anymore fulfilled (*i.e.* $\mathcal{O} = 0$). The permissiveness of the agent for the diversity ratio when closer to the objective $\mathcal{O}$ using the diversity adjustment factor (16) is modeled as:

$$f_\delta = 1 - \frac{\log(\eta) - \log(\eta_{\min})}{\log(\eta_{\text{init}}) - \log(\eta_{\min})} \tag{20}$$

while the empirically identified values in (14) and (15) are: $\eta_{\text{init}} = 1.5\,10^{-4}$, $\lambda_\eta = 50$, $\epsilon^*_{\delta,min} = 1$ and $\epsilon^*_{\delta,max} = 4$.

Simulation results, executed with a time horizon $T = 3$ and $N = 10$ action policies, are shown in Fig. 6. Baseline experiments with the 'random action agent' resulted in 6.1 steps per episode of in average for 500 episodes, with the best episode reaching 24 steps (in red in Fig.). As this environment starts with a pendulum at the goal state (the angle being near to 0), the 'no action agent' is a more interesting baseline, which lead to an average of steps per episode of 25.3 over 500 episodes, for which the agent was able to achieve only 2 successive episodes of 35 steps (in yellow in Fig.). The 'random generated action agent' failed systematically to outperform the 'no action agent' baseline (in magenta in Fig.). Agents using the diversity-based action choice strategy were able to solve the problem with notably better performance (between 100 and 600 steps per episode in average) than the 'no action agent'. The performance consistency through episodes was however much lower for the best performing agents (in blue and black in the Fig.). The latter were also able to reach several times the environment maximum number of episodes (10^3).

The Fig. 7 shows the phase diagrams corresponding to the 400 first steps of a successful episode of the best performing agent (blue in Fig. 6). The pendulum angle was maintained by the agent in a range below 25% of the maximum allowed angle, caused by a typical movement of the cart to achieve so. Fast changes of the action value can be seen, which oscillates around 0, showing the ability of the agent to react rapidly to avoid moving too far from the preferred prior.

VI. Discussion

The above results show that the DAIF agents generating actions as posteriors of hidden state transitions were able to learn the dynamics of both environments and solve the corresponding problems. The diversity-based strategy described in section III enhanced notably the agent performance when compared to the baseline agents. More specifically, using this strategy was the only way for the agents to steadily outperform the 'no action agent' baseline in the Inverted Pendulum problem, as the 'random generated action agent' systematically failed to achieve so.

The fact that the EFE ((12)) is reward-free, does not include empirical weighting parameter to balance risk and ambiguity components and that the training does not rely on expert experiments is one of the main strengths of this model as it requires less engineering effort and hyper-parameter tuning. The time-steps horizon and number of policies (2 and 10 respectively) used for the Mountain Car problem may be compared to typical 100 samples per action policy for at least 30 time-steps horizon for each time-step in [16]. This suggests that this action generating model coupled to the diversity-based, 'habit' action planning requires less computational effort dedicated to the planing task. Besides, the ability to restart the learning and recover stable performance shown in Inverted Pendulum results hints to a plasticity characteristic of the agent's model, being able to adapt to new situations where the agent beliefs do not correspond anymore to the observed situations.

One may notice that relying on the randomly sampled actions in generated set approach (performed by the 'random generated actions agent') lead to successful learning in the case where the problem dynamics do not require frequent, large gradients of actions as in the Mountain Car problem, but seemed to fail in providing enough model reactivity, as needed in the Inverted Pendulum problem.

In the scope of our experiences, for which the duration was limited in number of episodes or steps, the proportion of agents learning successfully the problem was about 10%. Integrating top-down attention was key to stabilize the agent behavior around a parameter subspace where the objective is achieved. This suggests that more work should be done on stabilizing the learning, by *e.g.* enhancing the top-down attention model, identifying a more stable optimization algorithm, or introduce efficient regularization techniques.

VII. Conclusion

We have shown in this paper how a deep active inference agent learning hidden state transitions to generate its actions can solve complex toy problems in continuous learning mode. We have also shown that agent's

action choice based on the generated actions' and EFE's diversities can be a viable and less costly alternative to already existing approaches. As a perspective for further investigations, the high-level agent could even be an Active Inference agent itself, learning to generate 'internal' actions, which may correspond in our study to the steering of top-down attention parameters, destined to steer the behavior of lower-level active inference agents. This work may open the door to applying DAIF to more realistic tasks where obtaining expert experiments or environment reward is difficult, where learning continuously is necessary as in environments constantly changing (*e.g.* robot or vehicle navigation) and where an integration of high-level and low-level DAIF agents could be key to solve multi-objective tasks.

REFERENCES

[1] K. Friston, J. Kilner, and L. Harrison, "A free energy principle for the brain," *Journal of Physiology Paris*, vol. 100, no. 1-3, pp. 70–87, 2006, arXiv: 1401.4122v2 ISBN: 0928-4257.

[2] K. J. Friston, T. Parr, Y. Yufik, N. Sajid, C. J. Price, and E. Holmes, "Generative models, linguistic communication and active inference," *Neuroscience & Biobehavioral Reviews*, vol. 118, pp. 42–64, Nov. 2020.

[3] T. Parr, G. Pezzulo, and K. J. Friston, *Active inference: the free energy principle in mind, brain, and behavior*. Cambridge, Massachusetts: The MIT Press, 2022.

[4] P. Mazzaglia, T. Verbelen, O. Çatal, and B. Dhoedt, "The Free Energy Principle for Perception and Action: A Deep Learning Perspective," *Entropy*, vol. 24, no. 2, p. 301, Feb. 2022, arXiv:2207.06415 [cs, q-bio].

[5] D. Foster, *Generative Deep Learning*. O'Reilly Media, Inc., 2019.

[6] K. J. Friston, J. Daunizeau, J. Kilner, and S. J. Kiebel, "Action and behavior: a free-energy formulation," *Biological Cybernetics*, vol. 102, no. 3, pp. 227–260, Mar. 2010.

[7] T. Isomura, K. Kotani, Y. Jimbo, and K. J. Friston, "Experimental validation of the free-energy principle with in vitro neural networks," *Nature Communications*, vol. 14, no. 1, p. 4547, Aug. 2023, number: 1 Publisher: Nature Publishing Group.

[8] K. Ueltzhöffer, "Deep active inference," *Biological Cybernetics*, vol. 112, no. 6, pp. 547–573, Dec. 2018.

[9] B. Millidge, "Deep Active Inference as Variational Policy Gradients," Jul. 2019, arXiv:1907.03876 [cs].

[10] O. Çatal, S. Wauthier, C. De Boom, T. Verbelen, and B. Dhoedt, "Learning Generative State Space Models for Active Inference," *Frontiers in Computational Neuroscience*, vol. 14, p. 574372, Nov. 2020, publisher: Frontiers Media S.A.

[11] Z. Fountas, N. Sajid, P. A. M. Mediano, and K. Friston, "Deep active inference agents using Monte-Carlo methods," in *34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada*, 2020, arXiv: 2007.05838.

[12] O. van der Himst and P. Lanillos, "Deep Active Inference for Partially Observable MDPs," in *Active Inference*, ser. Communications in Computer and Information Science, T. Verbelen, P. Lanillos, C. L. Buckley, and C. De Boom, Eds. Cham: Springer International Publishing, Dec. 2020, pp. 61–71.

[13] N. van Hoeffelen and P. Lanillos, "Deep Active Inference for Pixel-Based Discrete Control: Evaluation on the Car Racing Problem," Sep. 2021, arXiv:2109.04155 [cs].

[14] O. Çatal, J. Nauta, T. Verbelen, P. Simoens, and B. Dhoedt, "Bayesian policy selection using active inference," in *Workshop on "Structure & Priors in Reinforcement Learning" at ICLR 2019*, 2019, arXiv: 1904.08149v2.

[15] D. Silver and J. Veness, "Monte-Carlo planning in large POMDPs," in *Proceedings of the 23rd International Conference on Neural Information Processing Systems - Volume 2*, ser. NIPS'10. Red Hook, NY, USA: Curran Associates Inc., Dec. 2010, pp. 2164–2172.

[16] O. Çatal, T. Verbelen, J. Nauta, C. D. Boom, and B. Dhoedt, "Learning Perception and Planning With Deep Active Inference," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2020, pp. 3952–3956, iSSN: 2379-190X.

[17] A. Tschantz, M. Baltieri, A. K. Seth, and C. L. Buckley, "Scaling active inference," Nov. 2019, arXiv:1911.10601 [cs, eess, math, stat].

[18] B. Millidge, A. Tschantz, and C. L. Buckley, "Whence the Expected Free Energy?" *Neural Computation*, vol. 33, no. 2, pp. 447–482, Feb. 2021. [Online]. Available: https://doi.org/10.1162/neco_a_01354

[19] P. Schwartenbeck, T. FitzGerald, R. Dolan, and K. Friston, "Exploration, novelty, surprise, and free energy minimization," *Frontiers in Psychology*, vol. 4, Oct. 2013.

[20] K. Friston, "The free-energy principle: a rough guide to the brain?" *Trends in Cognitive Sciences*, vol. 13, no. 7, pp. 293–301, Jul. 2009.

[21] P. Dayan, S. Kakade, and P. R. Montague, "Learning and selective attention," *Nature Neuroscience*, vol. 3, no. S11, pp. 1218–1223, Nov. 2000.

[22] X. Wu, T. Wang, C. Liu, T. Wu, J. Jiang, D. Zhou, and J. Zhou, "Functions of Learning Rate in Adaptive Reward Learning," *Frontiers in Human Neuroscience*, vol. 11, Dec. 2017, publisher: Frontiers.

[23] J. B. Inglis, V. V. Valentin, and F. G. Ashby, "Modulation of Dopamine for Adaptive Learning: A Neurocomputational Model," *Computational brain & behavior*, vol. 4, no. 1, pp. 34–52, Mar. 2021.

[24] M. B. Mirza, R. A. Adams, K. Friston, and T. Parr, "Introducing a Bayesian model of selective attention based on active inference," *Scientific Reports*, vol. 9, no. 1, p. 13915, Sep. 2019, publisher: Nature Publishing Group.

[25] T. Parr and K. J. Friston, "Uncertainty, epistemics and active inference," *Journal of The Royal Society Interface*, vol. 14, no. 136, p. 20170376, Nov. 2017, publisher: Royal Society.

[26] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," 2013, arXiv: 1312.6114v10.

[27] M. Benhenda, "ChemGAN challenge for drug discovery: can AI reproduce natural chemical diversity?" Aug. 2017, arXiv:1708.08227 [cs, stat].

[28] D. Friedman and A. B. Dieng, "The Vendi Score: A Diversity Evaluation Metric for Machine Learning," Jul. 2023, arXiv:2210.02410 [cond-mat, stat].

[29] M. McCloskey and N. J. Cohen, "Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem," in *Psychology of Learning and Motivation*, G. H. Bower, Ed. Academic Press, Jan. 1989, vol. 24, pp. 109–165.

[30] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "OpenAI Gym," 2016.

[31] E. Todorov, T. Erez, and Y. Tassa, "MuJoCo: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Vilamoura-Algarve, Portugal: IEEE, Oct. 2012, pp. 5026–5033.