Reducing Intuitive-Physics Prediction Error through Playing

Olivier L. Georgeon[1,2][0000−0003−4883−8702], Béatrice de Montéra[1,3][0000−0002−3579−4870], and Paul Robertson[4][0000−0002−4477−0379]

[1] UR CONFLUENCE: Sciences et Humanites (EA 1598), UCLy, France
ogeorgeon@univ-catholyon.fr
bdemontera@univ-catholyon.fr
[2] SyCoSMA, LIRIS, CNRS, Villeurbanne, France
[3] MetaGenoPolis Unit, INRAE, Jouy-en-Josas, France
[4] DOLL Labs, Lexington, MA, USA
paulr@dollabs.com

Abstract. We present a mobile robot that autonomously generates behaviors to calibrate its intuitive-physics engine, also known as the "Game Engine in the Head" (GEITH). Most POMDP and Active Inference learning techniques operate in a closed world in which the set of states is defined a priori. However, implementing an "innate" GEITH and a set of interactive behaviors allowed us to avoid these limitations and design a mechanism for information search and learning in an open world. The results show that over a few tens of interaction cycles, the robot's prediction errors decrease, which shows an improvement in the GEITH calibration. Moreover, the robot generates behaviors that human observers describe as playful.

Keywords: Active inference · developmental learning · enaction · intrinsic motivation · robotics · core knowledge.

1 Introduction

In their book "The pragmatic Turn", Engel, Friston, and Kragic [2] advocate a shift from a representation-centered perspective to an "action-oriented" perspective on cognition. Aligned with this shift, we present a robotics implementation to study the intricacy between action and perception. Closely related to pragmatic philosophy, Whitehead's process philosophy proposes useful concepts to describe perception from entities having unconscious experience of connection to the world, which he calls *enduring objects* [23]. He distinguishes between two modes of perception: the mode of *presentational immediacy* and the mode of *causal efficiency* [17]. We apply these concepts to robotics: sensory input is provided to the robot in the mode of presentational immediacy, and the robot learns spatio-sequential patterns of actions and outcomes that account for the causal-efficiency mode of perception.

Karl Friston and his research group have proposed Active Inference as a method to interactively learn the causes of sensory signals by minimizing *free*

energy, or equivalently, prediction error [3]. Active inference has been used in the framework of Partially Observable Markov Decision Processes (POMDPs) to allow artificial agents to learn a causal model of the environment that they can only partially observe [18]. The causal model uses the distribution of the probability of each possible state of the world. In essence, at each instant, the agent estimates which states are the most or least likely to be the actual state of the world. The *expected information gained* to enhance this estimate is involved when selecting the next action. Active inference has been used in robotics [11] but generally under *closed-world* settings in the sense that the set of possible states is known *a priori*—a requirement for most of the mathematical apparatus of active inference.

When the robot is thrown into an open world, the problem of learning the cause of sensory input remains open. The POMDP and active inference literature suggest that the robot needs prior assumptions about the world to cope with complexity [7]. The present study examines how the "Game Engine In The head" (GEITH) can work as a suitable prior assumption that an autonomous robot could use to maintain a causal model of perception in the open world and reduce prediction errors.

Joshua Tenenbaum and his research group have proposed GEITH [21] as the capacity of cognitive beings to simulate the basic dynamics of physics and interactions. In the brain, the GEITH rests on structures that are partially predefined by genes and then completed through ontogenetic development. Similarly, it is possible to endow artificial agents and robots with a predefined software game engine and expect them to refine the parameters of their game engine and modify their predictions through ongoing interaction. The refinement of the game engine is assessed through the prediction error of sensory signals. The decrease in prediction errors shows an improvement of the game engine.

Compared with general Bayesian models classically used in active inference, GEITH adds the assumption that all the sensorimotor experiences can be localized in the 3D Euclidean space, at least approximately. It implements predefined linear-algebra functions to compute spatial transformations (rotations and translations) between frames of reference.

2 Our hypothesis

We comply with active inference theory in several regards. Firstly, we do not assume that sensory signals are *representational* of the state of the world. The world is hidden to the agent so that a given state may return contrary sensory signals when acted on differently by the agent. This implies a "conceptual inversion" of the interaction cycle in which action comes first and the sensory signal comes second as an *outcome* of action. Secondly, we do not provide the agent with presupposed ontological knowledge about entities in the world. The agent must infer the presence of *causes* in the world through patterns of interactive experience. Thirdly, no extrinsic goal is encoded in the agent in the form of goal states that the agent should search based on reward or other criteria. However,

we may associate some *prior preference* with interactions. In short, the agent has no *rewarding world states* but has *rewarding interactions*. For a deeper examination of these principles in relation to the active inference literature, we refer the reader to [7].

We also adopt *prediction error* as a measure of the quality of the agent's world model. However, we are not using the gradient descent of the prediction error as a motivational principle to drive the learning process. As we develop, our agent is not always driven by a value optimization process; it may also enact behaviors that we call *disinterested*. Disinterested behaviors can consist of aimless innate tendencies, or individual habits taken through the agent's lifetime. In this implementation, the reduction of prediction errors is not a means of improving the world model, but a consequence of its improvement.

We are using a cognitive architecture designed previously based on sensorimotor and enactive principles [7]. The present article reports the integration of the new GEITH module within this cognitive architecture, as illustrated in Figure 1. The GEITH supports the simulation of behaviors before their selection by the cognitive architecture and their enaction by the robot. At the beginning of each interaction cycle, the simulation computes the *predicted outcome*. At the end of the interaction cycle, the predicted outcome is compared with the *actual outcome* to calculate the prediction error. We investigate the core elements of the GEITH that are needed for the agent to reduce prediction error.

We draw inspiration from studies on *core knowledge* in the brains of animals and human infants. For example, Elizabeth Spelke and her colleagues argued for the existence of two core geometry systems that "evolved before the emergence of the human species": "The *core navigation system* captures absolute distance and sense [..] but not relative length or angle; the *core form analysis system* does the reverse" [19, p. 2789]. We start by implementing the minimal requirements she deems necessary for both systems, namely the ability to handle points in spatial memory, the foundational elements of Euclidean geometry.

Our cognitive architecture encodes behaviors as *composite interactions* which are sequences of *primitive interactions*. A primitive interaction is a *control loop* that involves actuator commands, expected sensory feedback, spatio-temporal attributes, termination conditions, termination outcome, and prior preference. Examples are given in Section 3. GEITH may consider some of the outcomes as the result of interaction with "something" in the environment. In this case, the GEITH instantiates a data structure called a *phenomenon*[5] and localizes this phenomenon at the position of the interaction in spatio-sequential memory. Next, GEITH simulates subsequent interactions with phenomena to predict future outcomes.

We seeded the cognitive architecture with "innate" composite interactions that cause the robot to explore the environment and interact with points encountered on the floor (Fig. 1, top center). In other studies, we implemented the

[5] Common-sense usage of the term *phenomenon*: "something" that a cognitive being perceives in the environment. Technically: "any useful grouping of a subset of spatio-temporal patterns experienced by an agent in an environment" [20, p. 8].

learning of new composite interactions [8], but here we only examine the refinement of the GEITH parameters allowing the tuning of primitive interactions to reduce the prediction error.

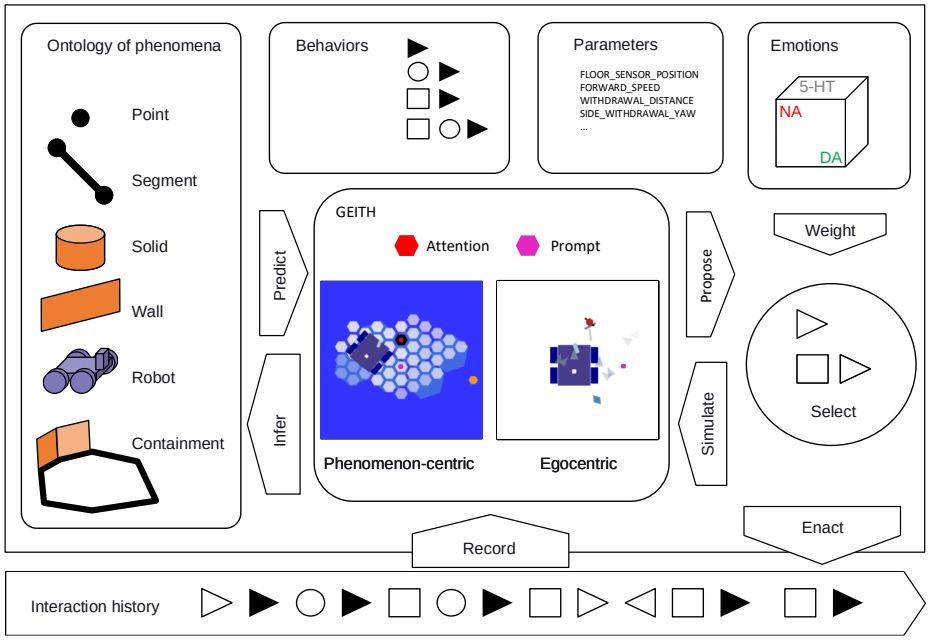


Fig. 1. The game engine within the cognitive architecture (derived from [7], Fig. 6). Bottom: the history of interactions enacted over time. Rightward triangles: `forward`. Leftward triangles: `backward`. Squares: `swipe`. Circles: `turn`. shade: outcome `white` or `black`. Center: the GEITH. Red hexagon: the focus of attention localized at the position of the phenomenon. Magenta hexagon: the prompt is the localization of the next selected interaction's destination: `swipe` to the right. Left: the types of phenomena inferred through interactive experience. Solid objects, walls, and other robots can be detected by the echo-localization sensor, but are not present in this experiment. Top center: predefined composite interactions and GEITH parameters. Top right: three-dimensional emotional state based on dopamine (DA), serotonin (5-HT), and nor-adrenaline (NA). Right: the cognitive architecture selects the next behavior based on the emotional state and the expected outcome predicted by the GEITH.

The cognitive architecture uses variables that represent the robot's *emotional state* to select composite interactions to try to enact. We use Hugo Lövheim's "cube of emotions" [12] as a basic emotional model based on three neurotransmitters: dopamine (DA), serotonin (5-HT), and nor-adrenaline (NA) (Fig. 1, top right). This model associates dopamine with pleasure and reward seeking behavior, serotonin with well-being and playful behavior, and nor-adrenaline with responses to arousal and stress. It has been used successfully for simple

emotional robotics. Our robot visually indicates its predominant neurotransmitter level using an intuitive color code studied by Max Talanov and his team: green for dopamine, white for serotonin, red for noradrenaline, and blue when all three neurotransmitter levels are low [1].

The GEITH implements two levels of spatio-sequential working memory: *egocentric* and *allo-phenomenon-centric* (Fig. 1, center). The interactions and displacements are received in the egocentric reference frame based on the position of sensors and translation speed given as GEITH parameters, and the yaw measured by the Inertial Measurement Unit (IMU) which plays a similar role as the vestibular system (Fig. 2, top right). When the robot encounters a new phenomenon, the GEITH instantiates a new allocentric reference frame centered on this phenomenon to track the displacement of the robot relative to this phenomenon (Fig. 2, bottom right). Comparable mechanisms of coordinate conversion have recently been found in the brain [22] and implemented in other software cognitive architectures [15]. Phenomenon-centric frames relate to Jeff Hawkins' thousand brain hypothesis [9], according to which the brain records thousands of small spatio-temporal models to memorize interactions with different kinds of object.

Once the robot has selected an object in the environment, its serotonin level increases, which triggers behaviors of interaction with this object to calibrate its GEITH parameters. Phenomenon-centric memory is discretized into a hexagonal grid inspired by grid cells in the entorhinal cortex [13]. The cognitive architecture uses this grid as a small finite discrete model in which to search for information and optimize it.

3 Experiment

We designed a mobile robotic platform called Petitcat[6] based on the *Osoyoo robot car* [14]. The experiment reported here uses only two sensors. The *floor luminosity sensor* is a bar of 5 infrared-reflective sensors directed to the floor. From this bar of sensors, we retrieve 4 possible signals: `none`, `left`, `front`, or `right` signaling the absence or relative position of a black tape present beneath them. The IMU measures the yaw during the enaction of interactions. Note that Petitcat cannot see the black tape from a distance. He has no camera, lidar, or odometer. What looks like eyes on his head is an ultrasonic echo-localization sensor not exploited in this experiment. The emotion indicator is an RGB LED (Fig. 2).

The C++ software running on the robot's Arduino board controls the enaction of primitive interactions. A personal computer implements the GEITH and the cognitive architecture that remote controls the robot via Wi-Fi. The cognitive architecture selects the primitive interaction to try to enact and sends it to the robot. The robot tries to enact it and sends the outcome back to the PC. The code is open source and shared online [6].

[6] Sections 3 and 4 personalize the robot by name and pronoun to enhance readability. We do not claim that he has a psychology or gender.

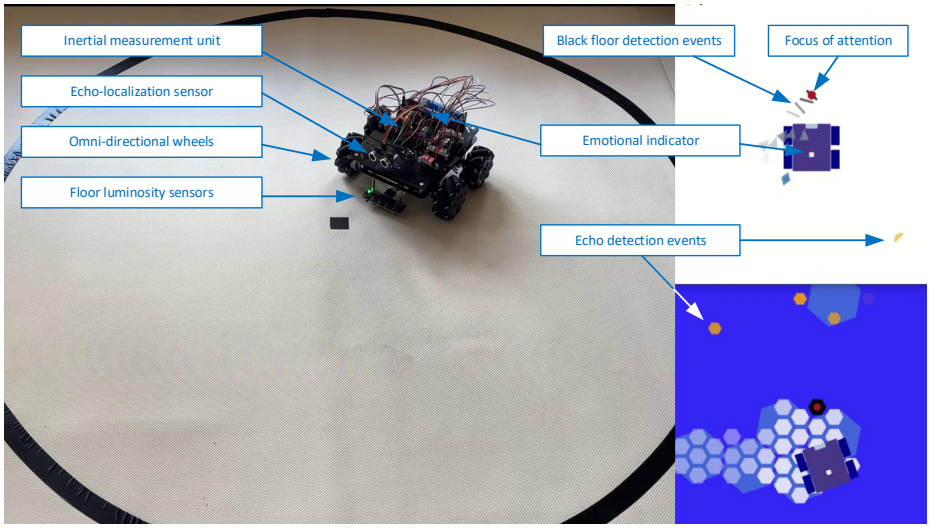


Fig. 2. Screenshot of a video example run [4]. Left: Petitcat playing with a point made of a piece of black tape on the floor. Top right: Petitcat's egocentric memory. Black segments: black tape detection events. Bottom right: phenomenon-centric memory. Black hexagon: the point phenomenon used as origin of the allocentric reference frame. Yellow hexagons: echo measured with the sonar. Red hexagon: focus of attention.

For this experiment, we defined four possible commands: `forward`, `backward`, `swipe`, and `turn`. `Forward` and `backward` are longitudinal translations. Their spatio-temporal attribute is the target duration (float). `Swipe` is a lateral translation. Its spatio-temporal attributes are the direction (left or right) and target duration (float). `Turn` consists in turning in place. Its spatio-temporal attribute is the target yaw (float), negative when counter-trigonometric.

The control loop monitors the elapsed time, yaw, and floor luminosity. The termination conditions are reaching the target duration or yaw, or detecting the black tape, making two possible outcomes: `white` or `black`. This gives eight primitive interactions identified by their tuple ⟨command, outcome⟩: 4 commands × 2 outcomes. All interactions are given a zero prior preference except ⟨`forward`, `white`⟩ which has a positive one. Additionally, the robot returns the measured spatio-temporal attributes: measured duration (float), measured yaw (float), and black tape detection (none, left, front, right).

When the black tape is detected, the movement is interrupted and a "reflex" movement is performed to withdraw from the tape by a few centimeters. When detection is on the side, this withdrawal includes a rotation to the opposite side, which tends to bring the robot back into a position perpendicular to the tape. This behavior was implemented to prevent the robot from falling off a table or exiting the arena.

We seeded the cognitive architecture with the four composite interactions below, which constitute "innate" ways for Petitcat to interact with points. The

GEITH tries to simulate them, computes their spatio-temporal attributes according to the position of the phenomenon in memory, and proposes those that are feasible in the current context.

1. ⟨⟨`forward`, `black`⟩⟩
2. ⟨⟨`turn`, `white`⟩, ⟨`forward`, `black`⟩⟩
3. ⟨⟨`swipe`, `white`⟩, ⟨`forward`, `black`⟩⟩
4. ⟨⟨`swipe`, `white`⟩, ⟨`turn`, `white`⟩, ⟨`forward`, `black`⟩⟩

Neurotransmitter levels can vary from 0 to 100 and are initialized at 50. DA prevails in case of equality. The prevalence of DA makes Petitcat initially select the ⟨`forward`, `white`⟩ interaction because it has a positive prior preference. When he detects a point (by surprise), 5-HT increases to its max. The prevalence of 5-HT and the presence of a point phenomenon in memory trigger the selection of innate interactions with the point. If the prediction errors do not decrease (that is, the prediction is not longer improving), 5-HT decreases. When 5-HT drops below or equal to DA, the ⟨`forward`, `white`⟩ interaction is again selected, causing Petitcat to explore new destinations.

Prediction errors may concern both the outcome of primitive interactions and the spatio-temporal measures. Prediction errors on the outcome (`black` predicted but `white` occurred, or the reverse) mean that the selected primitive interaction failed and another interaction was actually enacted instead. Failing primitive interactions cause the composite interaction to which they belong to abort, and NA to rise to its max.

The GEITH uses a *focus of attention* point and a *prompt* point to compute the spatio-temporal attributes of interactions (Fig. 1, center). When Petitcat interacts with a point, the GEITH places the focus of attention at the place of the phenomenon. A failure to interact with the point means that the localization of the phenomenon in memory is erroneous. The high NA level that occurs in case of failure causes the GEITH to move the focus of attention to another cell in phenomenon-centric memory in search of the point. Cells compete to catch the focus with preference given to those closer to the last detected position of the phenomenon but having gone the longest period since last being visited. The interactions with the point continue afterward based on the focus in different cells. NA is reset to 50 if Petitcat finds the lost point; otherwise, it progressively decreases until it drops below 50 causing Petitcat to abandon the search.

In addition to the number of failed interactions, we also expect the prediction errors of the yaw and of the forward duration to decrease as the GEITH adjusts its parameters. Our GEITH has about 20 parameters, but this experiment only involves `FLOOR_SENSOR_POSITION`, `FORWARD_SPEED`, `WITHDRAWAL_DISTANCE`, and `SIDE_WITHDRAWAL_YAW`. Note that GEITH has no means to infer the absolute values of these parameters but can only adjust them in relation to each other. The GEITH cannot either predict that the point will be detected on the side of the floor luminosity sensor, which will cause a withdrawal with rotation. The cognitive architecture makes the robot aim straight at the point. The GEITH thus always predicts a straight withdrawal.

4 Results

Several videos of experiment runs are available online [5]. Here we analyze the representative run recorded in [4]. In this run, Petitcat encountered the point in Step 1 and interacted with it up to Step 60. In Step 17, it missed the point, but found it again in Step 20. This is shown in the *outcome code prediction error plot* in Fig. 3. The fact that Peticat did not miss the point after Step 20 shows an improvement of the GEITH parameters.

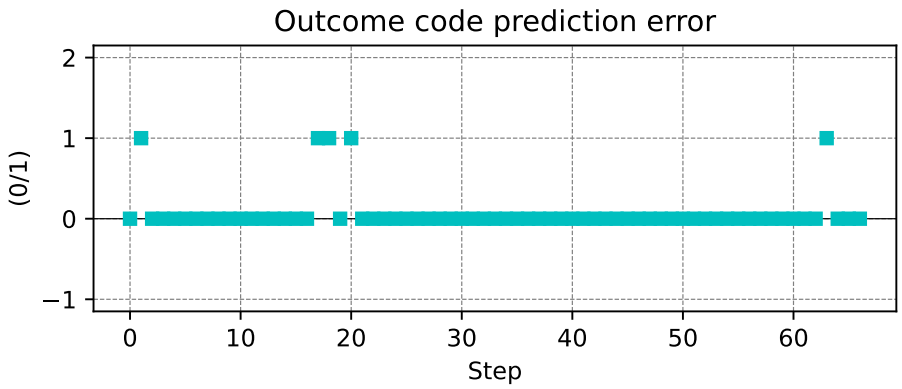


Fig. 3. Outcome prediction error plot: 0 if predicted outcome = actual outcome (successful interaction), 1 otherwise (failed interaction). Step 0: Petitcat moved forward. Step 1: he unexpectedly detected the point. Step 17: he expected to detect the point while translating forward but missed it. Step 18: he expected to not detect the point while turning but detected it. Step 20: he did not predict detecting the point but did. Step 63: As he moved away from the point, he did not expect to detect the arena border.

As explained above, the GEITH cannot predict when Peticat will detect the point on the side. This can cause large yaw prediction errors because the robot unexpectedly turned during withdrawal. Fig. 4 shows these prediction errors that do not improve over time.

The GEITH simulates turning while withdrawing based on the `SIDE_WITHDRAWAL_YAW` parameter. To adjust this parameter, the GEITH must compute the *yaw residual error* that is left when knowing on which side the point was detected. Fig. 5 shows that the residual yaw error decreases as the robot adjusts the `SIDE_WITHDRAWAL_YAW` parameter.

The adjustment of `FORWARD_SPEED` and `WITHDRAWAL_DISTANCE` allows for a visible decrease of the forward-duration prediction error shown in Fig. 6.

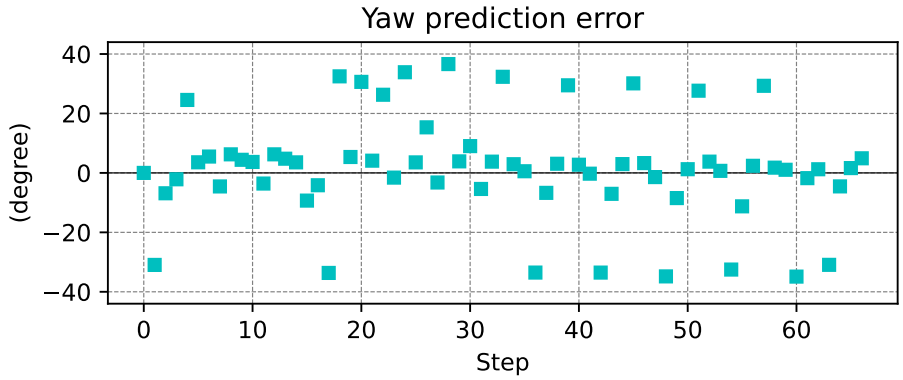


Fig. 4. Yaw prediction error plot. The prediction errors come from different causes which makes the interpretation of the plot difficult. Points above 20 or below -20 are large prediction errors occurring when Petitcat turned while withdrawing because the GEITH did not predict detecting the point on the side.

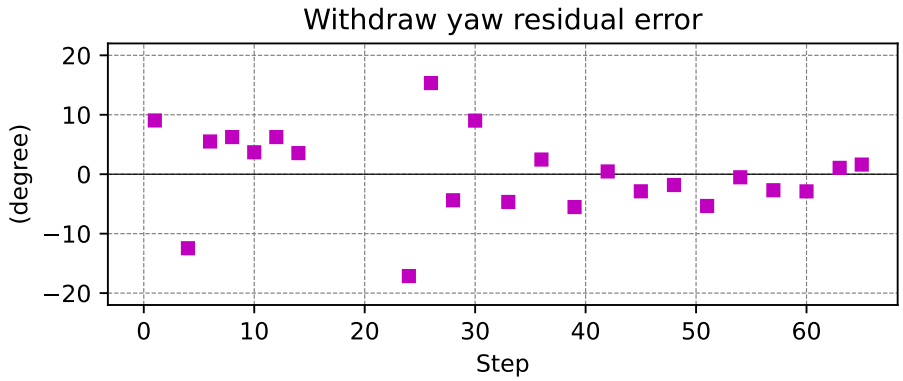


Fig. 5. Yaw residual error of interactions that have a `black` outcome. It shows a significant decrease as the robot interacts with the point.

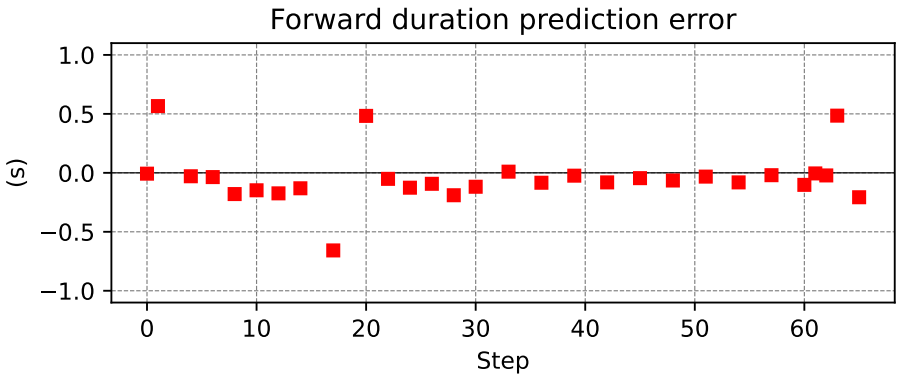


Fig. 6. Duration prediction error for `forward` interactions. Step 1 and 20: the forward translation was unexpectedly interrupted by the point detection. Step 17: the forward duration was longer than expected because the robot did not detect the point. Except for these events, the plot shows that the forward duration prediction error decreases as the robot is interacting with the point from Step 1 to 62. On Step 63: the robot moves away from the point and a forward duration prediction errors occur as he encounters a new object: the border of the arena.

5 Conclusion

We demonstrated a simple robot that managed to reduce intuitive-physics prediction errors in an open environment. We drew inspiration from theories positing core knowledge in the brain that have innate origins. Compared with our previous studies base on the same robot and earlier iterations of the cognitive architecture [7], the present study adds the model of emotion, the GEITH, and a set of innate behaviors. These elements are hard-coded in the robot, but what is not predefined is the set of world states and the ontology of objects in the world.

When the robot finds an object, it instantiates a small local model in the reference frame of this object. This finite discrete model lends itself to regular active inference techniques. We continue studying how to optimize the process of GEITH refinement in such local models using the active inference python library `inferactively-pymdp` [10]. This approach, however, remains dependent on the causal structure of the GEITH itself. How the robot could improve the causal structure of the GEITH or find exceptions remains an open question related to explainable AI [20].

This study illustrates Whitehead's two modes of perception. Mode 1, presentational immediacy (in the sense of "seeing a color"), corresponds to the robot's boolean sensory input we label `white` and `black`. Mode 2, causal efficiency (in the sense of "seeing an object"), corresponds to the robot maintaining a causal model of the point on the floor. We expect the next step to involve *mode-2 perception* of lines between points, which could open the way to learning the

compositionality of phenomena. We shall also examine how the robot can perceive the appearance, disappearance, and displacement of objects.

Beyond studying perception, we follow a method that advances through analogies between robots and natural organisms, aiming to enhance our understanding of the potential becoming of agency in robots. This method relates to the method of *transduction* proposed by another philosopher of process philosophy, Gilbert Simondon [16]. We try to design robots that mimic traits of natural organisms such as perception, surprise, emotions, and preferences. We are not claiming that the robot can actually *experience* these traits of agency, let alone have sentience. The robot, nonetheless, generates behaviors that human observers easily interpret as lifelike, which could find applications in companion robotics. The robot seems to enjoy exploring for the mere pleasure of movement as it lights up in green (DA prevails); it plays with the point as it does practicing its skills as it lights up in white (5-HT prevails); it seems anxious to search for the lost point as it lights up in red (NA prevails). This interpretation is also reinforced by seeing that the robot also turns its head in search of objects around the arena. In future experiments, we would like to study more precisely to what extent observers assign these subjective traits to the robot.

Acknowledgments. Dr. de Montera acknowledges support by ANR under contract ANR-11-DPBS-0001. Dr. Robertson acknowledges that this material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA), USA under Contract No. HR001120C0035.

References

1. Chebotareva, E., Safin, R., Shafikov, A., Masaev, D., Shaposhnikov, A., Shayakhmetov, I., Magid, E., Zilberman, N., Gerasimov, Y., Talanov, M.: Emotional social robot "emotico". In: 2019 12th International Conference on Developments in eSystems Engineering (DeSE). pp. 247–252. IEEE (2019). https://doi.org/10.1109/DeSE.2019.00054
2. Engel, A.K., Friston, K.J., Kragic, D.: The pragmatic turn: toward action-oriented views in cognitive science. Strüngmann forum reports, The MIT Press (2015)
3. Friston, K.: The free-energy principle: a unified brain theory? Nature Reviews Neuroscience **11**(2), 127–138 (2010). https://doi.org/10.1038/nrn2787
4. Georgeon, O.L.: Petitcat calibrates its intuitive physics engine (2024), `https://youtu.be/4wF-OeYCcYI`
5. Georgeon, O.L.: Petitcat playlist (2024), `https://youtube.com/playlist?list=PL1SPp5EpW5vFb-ZMCr8m0dIOoKEQe9CIE`
6. Georgeon, O.L.: Petitcat project repository (2024), `https://github.com/UCLy/INIT2/`
7. Georgeon, O.L., Lurie, D., Robertson, P.: Artificial enactive inference in three-dimensional world. Cognitive Systems Research **86**, 101234 (2024). https://doi.org/10.1016/j.cogsys.2024.101234
8. Georgeon, O.L., Riegler, A.: CASH only: Constitutive autonomy through motor-sensory self-programming. Cognitive Systems Research **58**, 366–374 (Dec 2019). https://doi.org/10.1016/j.cogsys.2019.08.006

9. Hawkins, J., Lewis, M., Klukas, M., Purdy, S., Ahmad, S.: A framework for intelligence and cortical function based on grid cells in the neocortex. Frontiers in Neural Circuits **12** (2019)
10. Heins, C., Millidge, B., Demekas, D., Klein, B., Friston, K., Couzin, I.D., Tschantz, A.: pymdp: A python library for active inference in discrete state spaces. Journal of Open Source Software **7**(73), 4098 (2022). https://doi.org/10.21105/joss.04098
11. Lanillos, P., Meo, C., Pezzato, C., Meera, A.A., Baioumy, M., Ohata, W., Tschantz, A., Millidge, B., Wisse, M., Buckley, C.L., Tani, J.: Active inference in robotics and artificial agents: Survey and challenges (2021). https://doi.org/10.48550/arXiv.2112.01871
12. Lövheim, H.: A new three-dimensional model for emotions and monoamine neurotransmitters. Medical Hypotheses **78**(2), 341–348 (2024). https://doi.org/10.1016/j.mehy.2011.11.016
13. Moser, E.I., Kropff, E., Moser, M.B.: Place cells, grid cells, and the brain's spatial representation system **31**(1), 69–89 (2008). https://doi.org/10.1146/annurev.neuro.31.061307.090723
14. Osoyoo: M2.0 metal chassis mecanum wheel robotic (2022), `https://osoyoo.com/2022/07/05/v2-metal-chassis-mecanum-wheel-robotic-for-arduino-mega2560-introduction-model-2021006600/`
15. Schneider, H.: The emergence of enhanced intelligence in a brain-inspired cognitive architecture **18** (2024). https://doi.org/10.3389/fncom.2024.1367712, publisher: Frontiers
16. Simondon, G., Garelli, J.: L'individuation à la lumière des notions de forme et d'information, ISBN: 9782841371815 Series: Krisis
17. Smith, O.B.: The social self of whitehead's organic philosophy. European Journal of Pragmatism and American Philosophy **II**(1). https://doi.org/10.4000/ejpap.935
18. Smith, R., Friston, K.J., Whyte, C.J.: A step-by-step tutorial on active inference and its application to empirical data. Journal of Mathematical Psychology **107**, 102632 (2022). https://doi.org/10.1016/j.jmp.2021.102632
19. Spelke, E.S., Lee, S.A.: Core systems of geometry in animal minds. Philosophical Transactions of the Royal Society B: Biological Sciences **367**(1603), 2784–2793 (2012). https://doi.org/10.1098/rstb.2012.0210
20. Thórisson, K.R.: The 'Explanation Hypothesis' in general self-supervised Learning. International Workshop in Self-Supervised Learning (2021)
21. Ullman, T.D., Spelke, E., Battaglia, P., Tenenbaum, J.B.: Mind games: Game engines as an architecture for intuitive physics. Trends in Cognitive Sciences **21**(9), 649–665 (2017). https://doi.org/10.1016/j.tics.2017.05.012
22. Wang, C., Chen, X., Knierim, J.J.: Egocentric and allocentric representations of space in the rodent brain. Current Opinion in Neurobiology **60**, 12–20 (2020). https://doi.org/10.1016/j.conb.2019.11.005
23. Whitehead, A.N.: Process and Reality: Corrected Edition. Free Press, New York (1978), David Ray Griffin & Donald W. Sherburne ed., 1929

Selection of Exploratory or Goal-Directed Behavior by a Physical Robot Implementing Deep Active Inference

Ko Igari, Kentaro Fujii, Gabriel W. Haddon-Hill, and Shingo Murata

Keio University, Japan
{koigari1222, oakwood.n14.4sp, gabe.haddon-hill}@keio.jp,
murata@elec.keio.ac.jp

Abstract. Intelligent robots are being developed with the expectation that they will perform various tasks in diverse environments. Such robots need to autonomously engage in both exploratory behavior to reduce environmental uncertainty and goal-directed behavior to achieve their preferred observations (or goals). In this study, we focus on active inference, which provides a unified scheme for these distinct behavioral modes. Policy selection in active inference is based on minimizing expected free energy (EFE), which consists of one term representing epistemic value and another representing extrinsic value. Specifically, we investigate the influence of preference precision, which controls the balance between these two terms, on policy selection by a physical robot receiving high-dimensional and uncertain observations. We developed a deep active inference framework comprising a world model and a policy suggester. The world model predicts future hidden states and observations based on candidate policies from the policy suggester. The EFE for each policy is approximated using the predicted future hidden states and observations as well as the preferred observation. We implemented our proposed framework in a robot, requiring it to select a policy that minimizes EFE and then generate actions accordingly. The experimental results showed that the robot implementing the proposed framework selected exploratory or goal-directed behavior depending on the level of preference precision. These findings suggest that adjusting preference precision plays a crucial role in the autonomous selection of exploratory or goal-directed behavior in real-world situations with potential uncertainty.

Keywords: Deep Active Inference · Free Energy Principle · World Model · Robot Learning

1 Introduction

Performing a wide range of tasks within diverse environments is a core goal of intelligent robots. Given that real-world situations often involve potential uncertainty, including unknown internal object properties and occlusions, these

robots must autonomously select exploratory or goal-directed behavior based on the situation. More specifically, exploratory behavior should be selected to reduce environmental uncertainty, whereas goal-directed behavior should be selected to achieve preferred observations (or goals). These distinct behavioral modes have recently been studied in unsupervised reinforcement learning (RL) and active inference schemes.

Unsupervised RL [4, 13, 17, 20, 22, 23, 28] first pre-trains agents with intrinsic rewards via exploration and then fine-tunes them with extrinsic rewards for adapting to downstream tasks or achieving goals. Although unsupervised RL agents can efficiently adapt to downstream tasks, manually designing intrinsic and extrinsic rewards is a challenging task [24, 25]. Moreover, the pre-training phase with exploratory behavior and the fine-tuning phase with goal-directed behavior are explicitly separated. In contrast, active inference based on the free energy principle (FEP) [6] more naturally promotes the distinct behavioral modes in terms of minimizing the expected free energy (EFE) [8]. The EFE consists of one term representing epistemic value and another representing extrinsic value. Because the former encourages exploratory behavior and the latter encourages goal-directed behavior, achieving balance between these two terms is important for autonomous policy selection.

One crucial issue in active inference is that most previous studies have been limited to toy problems such as a T-maze task in low-dimensional discrete environments [7, 8]. To overcome this limitation, deep active inference leverages deep neural networks (DNNs) to represent the recognition and generative models of active inference [18, 31]. This enables active inference to work in high-dimensional continuous environments. However, applications of deep active inference have mostly been limited to simulated environments [5, 12, 19, 27, 30]. Although some studies have applied deep active inference in physical robots for body perception and action [26], autonomous navigation [1, 2], and object manipulation [14, 16], the experimental setups and policy variations they employed were relatively simple. Moreover, to our knowledge, only the study by Çatal et al. [2] has considered the selection of exploratory or goal-directed behavior.

To tackle these issues, we aim to build upon previous research that integrated physical robots and deep active inference. Specifically, we consider robotic object manipulation and the selection of exploratory or goal-directed behavior in complex real-world environments. To this end, we develop a deep active inference framework consisting of a world model for handling high-dimensional and uncertain observations as well as a policy suggester for reducing the number of candidate policies. By using the world model and policy candidates from the policy suggester, we approximate the EFE for each policy. Importantly, in the EFE, we introduce *preference precision*, which controls the balance between the epistemic value for exploratory behavior and the extrinsic value for goal-directed behavior. We evaluate the robot's behavior depending on the preference precision in real-world environments with multiple manipulatable objects.

2 Active Inference

Active inference is a theoretical framework based on the principle that biological agents including humans minimize free energy to adapt to their environment [6]. This framework provides a compelling and unifying account of how biological agents manage perception and action to reduce surprise associated with their external environment.

Free Energy Principle The FEP posits that sensory observations o arise from hidden states s via a generative process, and that agents use a generative model $p(o, s)$ to infer these states. According to the FEP, agents aim to minimize the (Shannon) surprise of observations, represented as $-\ln p(o_t)$, where t denotes the current time step. However, because it is difficult to directly compute surprise, variational free energy F, which is the upper bound on surprise, is used by introducing an approximate posterior or recognition model of the hidden states $q(s_t)$. The variational free energy at the current time step (F_t) is defined as follows:

$$\begin{aligned} -\ln p(o_t) &= -\ln \int q(s_t) \frac{p(o_t, s_t)}{q(s_t)}\, ds_t \\ &\leq -\mathbb{E}_{q(s_t)}\left[\ln \frac{p(o_t, s_t)}{q(s_t)}\right] \\ &= \underbrace{D_{\mathrm{KL}}\left[q\left(s_t\right) \parallel p\left(s_t\right)\right]}_{\text{Complexity}} - \underbrace{\mathbb{E}_{q(s_t)}\left[\ln p\left(o_t \mid s_t\right)\right]}_{\text{Accuracy}} \\ &\triangleq F_t\,. \end{aligned} \tag{1}$$

Expected Free Energy In the active inference framework, agents select a policy that minimizes EFE. The EFE at a future time step τ with respect to policy π ($G_\tau(\pi)$) is described as follows:

$$G_\tau(\pi) = -\underbrace{\mathbb{E}_{q(o_\tau|\pi)}[D_{\mathrm{KL}}[q(s_\tau \mid o_\tau, \pi) \| q(s_\tau \mid \pi)]]}_{\text{Epistemic Value}} - \underbrace{\mathbb{E}_{q(o_\tau|\pi)}[\ln p(o_\tau \mid C)]}_{\text{Extrinsic Value}}\,. \tag{2}$$

The first term represents the epistemic value, which quantifies the reduction in uncertainty about the hidden states s_τ by receiving the observations o_τ. The second term represents the extrinsic value, which measures how well the observations o_τ align with the preferred condition C (e.g., a goal image o_{g}).

For policy selection, agents calculate the EFE for all policies within a defined planning horizon. The distribution over policies is represented by $p(\pi) = \sigma(-G(\pi))$, where σ denotes the softmax function and $G(\pi)$ denotes the temporal sum of $G_\tau(\pi)$. By choosing a policy that minimizes EFE, agents can effectively navigate the trade-off between maximizing informational gain and achieving a particular goal.

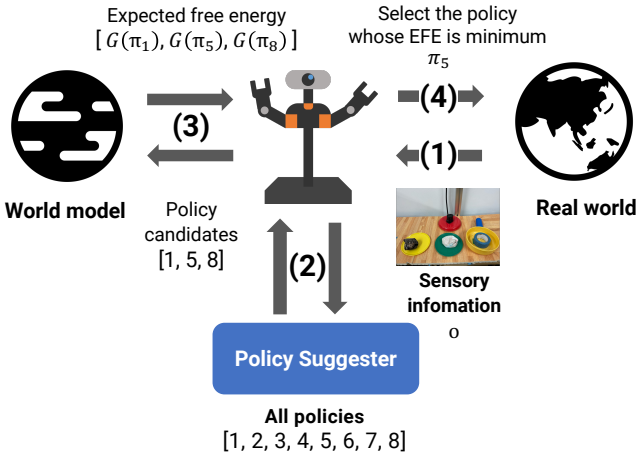


Fig. 1. Policy–selection steps in our proposed framework. (1) A robot acquires an initial observation from a camera mounted on it. (2) A policy suggester receives this observation as input and outputs policy candidates. (3) A world model is used to calculate EFE for each policy candidate. (4) The policy with the lowest EFE is selected, and the robot acts in the real world based on this policy.

3 Methods

Our proposed framework utilizes two DNN-based models for policy selection: a world model and a policy suggester. Policy selection is performed as follows(see Fig. 1). First, the robot acquires an initial observation from a camera mounted on it. Second, the policy suggester receives this observation as input and outputs policy candidates. Third, the world model is used to calculate the EFE for each policy candidate. Finally, the policy with the lowest EFE is selected, and the robot acts in the real world based on this policy.

3.1 Models

World Model World models [3, 9, 21, 29] represent the environment's state transitions, which are influenced by an agent's actions, thereby enabling the agent to predict future observations. In active inference, this model, which includes an encoder and a decoder resembling a variational autoencoder [15], is part of a partially observable Markov decision process framework.

At a given time step t, the agent acquires an observation o_t and infers the hidden state z_{t-1}. The agent then selects an action a_{t-1}, which is used by the transition model to predict the next state z_t and acquire the next observation o_t. This study employs a recurrent state space model (RSSM) [10] that handles both the deterministic states d_t and stochastic states s_t for the hidden states $z_t = \{d_t, s_t\}$. Thanks to this explicit separation of the hidden states to deterministic and stochastic parts, the RSSM effectively represents and predicts

the environment's dynamics by combining both the predictable and uncertain aspects.

The RSSM is defined as follows:

$$\text{RSSM} \begin{cases} \text{Deterministic State:} & d_t = f_\phi\left(z_{t-1}, a_{t-1}\right) \\ \text{Approximate Posterior:} & s_t \sim q_\phi\left(s_t \mid d_t, o_t\right) \\ \text{Prior:} & s_t \sim p_\phi\left(s_t \mid d_t\right) \\ \text{Likelihood:} & o_t \sim p_\phi\left(o_t \mid z_t\right). \end{cases} \tag{3}$$

With reference to the DreamerV2 architecture [11], we employed the gated recurrent unit for the deterministic function f_ϕ and assumed categorical distributions for the approximate posterior and prior, as well as the Gaussian for the likelihood. The approximate posterior and likelihood were implemented by convolutional neural network–based encoder and decoder architectures, respectively. The prior was implemented by a multilayer perceptron–based architecture. The essential point is that the approximate posterior or recognition model receives the current observation while the prior or transition model does not receive this observation. By training these two distributions to be close to each other, the RSSM can generate future hidden states by using the prior without receiving observations. This generation method is called latent imagination [10] and it is utilized to compute EFE as detailed later.

The RSSM is trained to minimize the temporal summation of the following variational free energy as follows:

$$F_t = D_{\mathrm{KL}}\left[q_\phi\left(s_t \mid d_t, o_t\right) \parallel p_\phi\left(s_t \mid d_t\right)\right] - \mathbb{E}_{q_\phi(s_t \mid d_t, o_t)}\left[\ln p_\phi\left(o_t \mid z_t\right)\right]. \tag{4}$$

Policy Suggester Because not all policies are executable for every situation, we employed an additional model called the policy suggester f_ψ, which suggests only the executable policies in a given situation. By introducing this model, we can efficiently calculate and evaluate the EFE only for the executable policies.

The policy suggester takes an observation o as input and predicts executable policies as multi-labels in the following manner:

$$\hat{y} = f_\psi(o), \tag{5}$$

where the ith element of $\hat{y}$ ($\hat{y}_i$) represents the predicted probability for the ith policy π_i. The policy suggester is trained to minimize the binary cross-entropy loss for each policy, similar to conventional multi-label classification.

3.2 Expected Free Energy

Approximation of EFE The strict calculation of the original EFE $G_\tau(\pi)$ in (2) is difficult due to its expectation operation ($\mathbb{E}_{q(o_\tau \mid \pi)}[\cdot]$) for the epistemic and extrinsic values. Therefore, we approximate the original EFE by introducing a sampling-based Monte Carlo method as detailed below.

The epistemic value in (2) is approximated by the empirical mean of the Kullback–Leibler (KL) divergence between the approximate posterior $q(s_\tau \mid o_\tau^{i,j}, \pi)$ and the prior $q(s_\tau \mid \pi)$ across $M \times N$ sampled observations as follows:

$$\text{Epistemic Value: } \frac{1}{N}\frac{1}{M}\sum_{j=1}^{N}\sum_{i=1}^{M} D_{\mathrm{KL}}\left[q\left(s_\tau \mid o_\tau^{i,j}, \pi\right) \| q\left(s_\tau \mid \pi\right)\right] \tag{6}$$

where $o_\tau^{i,j}$ are samples from $p\left(o_\tau \mid s_\tau^j\right)$, and M and N are the number of sampled states and observations, respectively. The extrinsic value in (2) is approximated by the empirical mean of the log likelihood $\ln p(o_\tau^{i,j} \mid C)$ across $M \times N$ sampled observations as follows:

$$\text{Extrinsic Value: } \frac{1}{N}\frac{1}{M}\sum_{j=1}^{N}\sum_{i=1}^{M} \ln p(o_\tau^{i,j} \mid C), \tag{7}$$

where C represents the preferred condition for the observations. Finally, the approximated EFE is given as the sum of the negative epistemic and extrinsic values.

Calculation of EFE Approximation It is necessary for the approximate EFE to be calculated by the RSSM components defined in (3). For this, we first establish an approximate posterior $q(s_1 \mid o_1, \pi) = q_\phi(s_1 \mid d_1, o_1)$ from an initial observation o_1. We use the M hidden states $\{s_1^i\}_{i=1}^M$ sampled from this with policy $\pi = \{a_\tau\}_{\tau=1}^{T-1}$ to predict future hidden states $\{\hat{s}_\tau^i\}_{\tau=2}^T$ via latent imagination by using $q(s_\tau \mid \pi) = p_\phi(s_\tau \mid d_\tau^i)$, where $d_\tau^i = f_\phi(z_{\tau-1}^i, a_{\tau-1})$ and $z_{\tau-1}^i = \{d_{\tau-1}^i, \hat{s}_{\tau-1}^i\}$. The decoder $p_\phi(o_\tau \mid z_\tau^i)$ then samples N predicted observations $\{\hat{o}_\tau^{i,j}\}_{j=1}^N$. These facilitate the creation of an approximate posterior $q(s_\tau \mid \hat{o}_\tau^{i,j}, \pi) = q_\phi(s_\tau \mid d_\tau^i, \hat{o}_\tau^{i,j})$ for each $\hat{o}_\tau^{i,j}$. This calculation is illustrated in detail in Fig. 2

Using the RSSM components, the epistemic value in (6) is calculated as the empirical mean of the KL divergence between the approximate posterior $q_\phi(s_\tau \mid d_\tau^i, \hat{o}_\tau^{i,j})$ and prior $p_\phi(s_\tau \mid d_\tau^i)$ across $M \times N$ sampled observations. We assume that the likelihood in (7) follows Gaussian $\mathcal{N}(o_{\mathrm{g}}, 1/\gamma)$ with mean o_{g} and variance $1/\gamma$. Here, the inverse variance corresponds to the preference precision, which controls the balance between the epistemic and extrinsic values. Given this assumption, the extrinsic value can be calculated as the mean squared error (MSE) between the goal observation o_{g} and predicted observation from the world model, scaled by the preference precision γ. Finally, the EFE approximation in this study is calculated as follows:

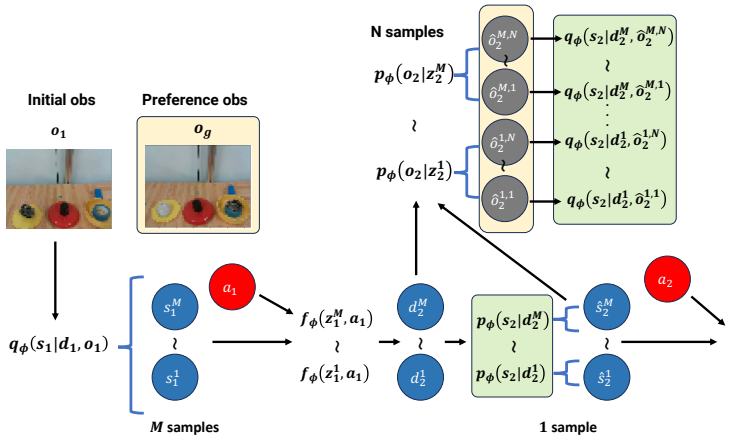


Fig. 2. Calculation of the approximate EFE. We first compute $q_\phi(s_1 \mid d_1, o_1)$ from o_1 and sample $\{s_1^i\}_{i=1}^M$. From these and a_1, $\{d_2^i\}_{i=1}^M$ are computed. We then compute $\{p_\phi(s_2 \mid d_2^i)\}_{i=1}^M$ and sample $\{\hat{s}_2^i\}_{i=1}^M$. Using $\{z_2^i\}_{i=1}^M$, we compute $\{p_\phi(o_2 \mid z_2^i)\}_{i=1}^M$ and sample $\{\hat{o}_2^{i,j}\}_{j=1}^N$. Finally, we compute $\{q_\phi(s_2 \mid d_2^i, \hat{o}_2^{i,j})\}_{j=1}^N$.

$$\text{EFE Approximation: } \underbrace{-\frac{1}{N}\frac{1}{M}\sum_{j=1}^{N}\sum_{i=1}^{M} D_{\mathrm{KL}}\left[q_\phi\left(s_\tau \mid d_\tau^i, o_\tau^{i,j}\right) \| p_\phi\left(s_\tau \mid d_\tau^i\right)\right]}_{\text{Epistemic Value}} \underbrace{-\frac{1}{N}\frac{1}{M}\sum_{j=1}^{N}\sum_{i=1}^{M} \gamma \cdot \left(-\mathrm{MSE}(o_{\mathrm{g}}, o_\tau^{i,j})\right)}_{\text{Extrinsic Value}}. \tag{8}$$

When calculating the extrinsic value, we focus on only the last K time steps of the planning horizon T due to the potential divergence between earlier predicted observations from the world model and the goal observation. Unexpected changes, such as the robot's arm entering the visual field, can cause deviations from the goal observation. Concentrating on the last time steps enables the environmental state to be more accurately captured post-policy execution. This ensures that the extrinsic values accurately reflect the success of the policy in achieving the intended goal. We rescale this extrinsic value for the original planning horizon by multiplying $\frac{T}{K}$.

4 Experiments

4.1 Experimental Setup

Robot In the experiments, we used the Rakuda-2 robot (Robotis Japan), shown in Fig. 3 (left), which has a total of 17 degrees of freedom (DOF), but two of the

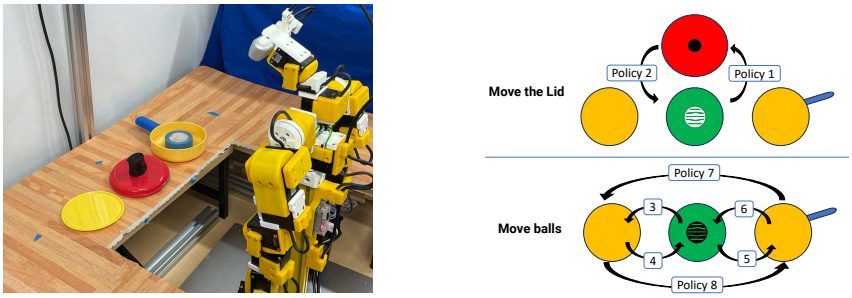


Fig. 3. Experimental environment (left) and the robot's policies (right). The environment consists of a dual-arm robot (Rakuda-2), yellow and green plates, a yellow pot, a red lid, and a black ball and/or a white ball. Each policy is a sequence of actions that involves moving either the lid or one of the balls from one position to another.

head joints were fixed. The robot head is equipped with a camera (RealSense Depth Camera D435; Intel) capable of capturing RGB images at a standard resolution of 480×640 pixels.

Task and Environment To demonstrate the scalability of the proposed framework with a physical robot, we designed object–manipulation tasks in a complex real-world environment. In our experimental setup (left in Fig. 3), the robot faced a table on which there were yellow and green plates, a yellow pot, and a red lid. In addition, the environment contained a black ball and/or a white ball on the plates or in the pot. Note that when the lid was on the green plate, the situation was uncertain because there might or might not be a ball on the plate. The robot was allowed to move either the lid or one of the balls from one position to another based on one of the eight predetermined policies (from π_1 to π_8) shown in Fig. 3 (right).

Dataset We collected visuo-proprioceptive data across three settings–each involving either a black ball, a white ball, or both. For each of these settings, five sets of data were collected by executing the predetermined policies, resulting in a total of 15 sets of data. During data collection, only meaningful policies were randomly selected within a given context. For instance, if the lid was already on the green plate, we did not execute the policy to put the lid on the green plate. The collected data included the robot's joint angles as well as visual images captured by the camera, which were recorded for 1000 time steps at a frequency of 5 Hz. In this experiment, the resolution of the visual images was downsized to 48×64 as observation o_t. The actions a_t were replaced with the 15-DOF joint angles j_t. The pixel values in the images were normalized within the range of 0 to 1, and the joint angles were normalized within the range of -0.95 to 0.95.

For the training of the world model, we created a dataset by randomly extracting 1200 sub-sequences with 50 time steps each from the collected data with 1000 time steps–300 sequences for the setting with only a white ball, 300 sequences for the setting with only a black ball, and 600 sequences for the setting with both balls. Due to these three settings, the world model was not able to determine whether there would be a white ball, a black ball, or no ball when the lid was on the green plate. Additionally, in these three settings, there were 19 possible environmental situations corresponding to the positions of the balls and the lid.

To train the policy suggester, we labeled 30 images of each situation with eight classes corresponding to the number of policies by assigning 1 for executable policies and 0 for non-executable policies in each situation. As a result, we created a dataset with a total of $19 \times 30 = 570$ images and corresponding labels.

4.2 Evaluation of Proposed Framework

Latent Imagination in Uncertain Scenarios To evaluate the ability of the world model to represent uncertainty in the environment, we conducted a latent imagination test starting from an uncertain situation. It was assumed that if the world model learned the uncertainty of the environment, it could predict different future observations from the same initial observation and policy.

The world model was first given an initial observation and sampled initial hidden states. Then, the world model predicted future hidden states with a particular policy via latent imagination from the initial hidden states. Finally, the world model predicted future observations with the decoder from the future hidden states.

Policy Selection in Uncertain Scenarios To evaluate how the robot's behavior changes depending on the preference precision, we performed a policy selection test starting from an uncertain situation. We expected that the robot would select a policy for exploratory behavior with low preference precision and a policy for goal-directed behavior with high preference precision.

In the experiment, an initial observation was first given to the policy suggester, and then only the policies with a probability from the suggester $\hat{y}$ higher than a threshold of 0.7 were used to calculate the approximate EFE. Finally, the approximate EFE was calculated with respect to the suggested policies and the robot selected the policy whose EFE was lowest. Here, we compared different precision settings $\gamma = \frac{1}{20}, \frac{1}{10}, 1, 3, 10, 20$, and conducted five trials in each setting. The other hyperparameters were $M = 50$, $N = 4$, $T = 40$, $K = 5$.

5 Results and Discussion

5.1 Latent Imagination in Uncertain Scenarios

The world model was required to predict future observations based solely on an initial observation and a particular policy via latent imagination. As an example,

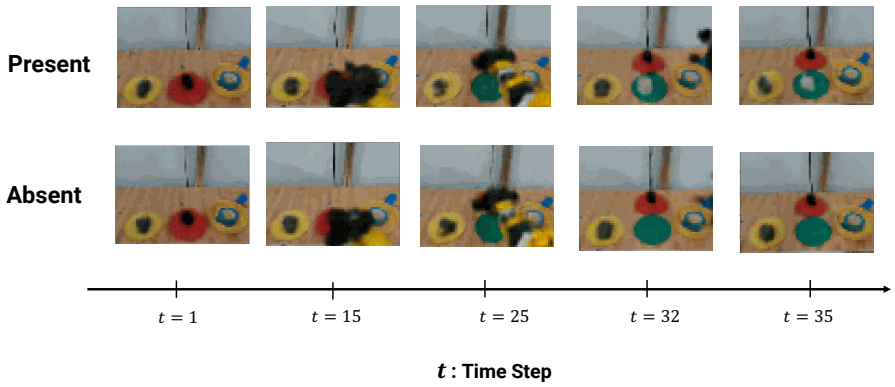


Fig. 4. Latent imagination results in an uncertain scenario. From the initial observation, it is unclear whether the environment contains the white ball under the red lid. The top row shows the result when the policy π_1 to open the lid was used for latent imagination, resulting in the appearance of the white ball on the green plate in the center. The bottom row shows the result with the same initial observation and policy, but without the appearance of the white ball on the green plate.

Fig. 4 shows the results with an initial observation featuring a black ball on the left yellow plate and a red lid in the center, and a policy π_1 for opening the lid. We sampled five sequences of the predicted observations with the same initial observation and policy, two of which are shown in the figure. One sequence demonstrates the green plate in the center with the white ball, while the other sequence shows it without the white ball.

These different predicted observations demonstrate that the stochastic states in the world model can represent uncertain situations, such as whether the green plate contains the white ball under the red lid or not.

5.2 Policy Selection in Uncertain Scenarios

The robot was required to select a policy based on the minimization of EFE. As an example, the same situation as in the latent imagination test was considered, with an initial observation featuring a black ball on the left plate and a red pot lid in the center. Additionally, an image featuring the black ball in the right pot and the red lid in the center was provided to the robot as the preferred observation. In this experiment, the preference precision was varied between $\frac{1}{20}$ and 20, as specified in Sec 4.2, with each setting being tested five times.

Figure 5 shows snapshots of the robot's actions generated based on the selected policy with the lowest and highest preference precision settings ($\gamma = \frac{1}{20}$ and $\gamma = 20$). In the lowest precision setting, the robot selected policy π_1, which involved opening the red lid. This behavior revealed whether the white ball was absent (top row in the figure) or present (middle row in the figure) on the green plate in the center. In contrast, in the highest precision setting, the robot se-

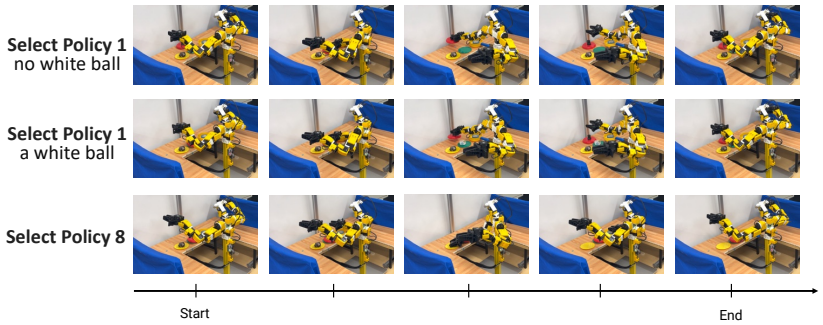


Fig. 5. Snapshots during the robot's actions generated based on the selected policy with different preference precision settings ($\gamma = \frac{1}{20}$ for the top and middle rows and $\gamma = 20$ for the bottom row). The top row shows the case where policy π_1, which involves opening the red lid in the center, was selected and there was no white ball on the green plate. The middle row shows the case where the same policy π_1 was selected, but there was a white ball on the green plate. The bottom row shows the case where policy π_8, which involves moving the black ball on the left plate to the right pot, was selected.

lected policy π_8, which moved the black ball from the left yellow plate to the right yellow pot. The resultant observation matched the preferred observation.

These results indicate that when the preference precision is low, the robot selects a policy that produces exploratory behavior aimed at reducing environmental uncertainty, temporarily ignoring preference satisfaction. In contrast, when preference precision is high, the robot selects a policy that produces goal-directed behavior aimed at achieving the preferred observation, temporarily ignoring environmental uncertainty.

To analyze the relationship between preference precision and the EFE with different policies, we calculated the difference between the EFEs of policy π_1 (opening the pot lid) and those of policy π_8 (moving the ball from the left plate to the right pot), depending on preference precision γ. Negative and positive values for this difference indicate that policies π_1 and π_8 are selected, respectively. As shown in Fig. 6, at lower γ values, the differences are negative, indicating that policy π_1 or exploratory behavior was selected. This is because when preference precision is low, the EFE is dominated by the epistemic value, which can be maximized (its negative minimized) by exploratory behavior. In contrast, increasing γ enhances the influence of the extrinsic value of the EFE, indicating that the goal-directed behavior of policy π_8 can minimize the EFE by maximizing the extrinsic value. Additionally, when the preference precision was $\gamma = 2$, the EFE difference between the two policies approached zero. At this value, the EFE of both policies varied between trials due to the probabilistic nature of the world model. In this case, policy π_1 was selected twice, while policy π_8 was selected three times across five trials.

These experimental results demonstrate that when the robot implements the proposed framework, it selects exploratory or goal-directed behavior depending

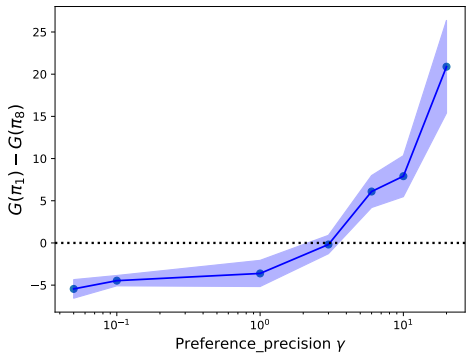


Fig. 6. Difference in EFE between two policies when the preference precision is varied. The horizontal axis represents the preference precision, and the vertical axis shows the EFE difference between policy π_1 (opening the pot lid) and π_8 (moving the ball from the left to the right pot) , denoted as $G(\pi_1) - G(\pi_8)$. Solid points indicate the mean across five trials, with shaded areas representing the standard deviation.

on the level of preference precision in the EFE. Moreover, adjusting the preference precision play a crucial role in the autonomous selection of exploratory or goal-directed behavior in real-world situations with potential uncertainty.

6 Conclusion

This study proposed a deep active inference framework consisting of a world model and a policy suggester with the aim of investigating the selection of exploratory or goal-directed behavior by a physical robot. Using the proposed framework, we calculated the approximate EFE, which includes an epistemic value term for engaging in exploratory behavior and an extrinsic value term for engaging in goal-directed behavior. By considering the preference precision in the EFE, the balance between these two terms was controlled. The experimental results for the object–manipulation task demonstrated that when the robot implemented the proposed deep active inference framework, selected exploratory behavior aimed at reducing environmental uncertainty when the preference precision was relatively low. In contrast, the robot selected goal-directed behavior aimed at achieving its preferred observation when the preference precision was relatively high. These findings suggest that the selection of distinct behavioral modes can be realized by attempting to minimize the EFE.

In future work, we plan to introduce a policy network, instead of predetermined policies, to enable adaptive behavior that is robust to environmental changes. We also plan to tackle more challenging tasks that require the robot to first select exploratory behavior in order to understand the environment and then select goal-directed behavior to achieve a particular goal.

Acknowledgement This work was supported in part by the Japan Science and Technology Agency (PRESTO Grant No. JPMJPR22C9) and the Japan Society for the Promotion of Science (KAKENHI Grant Number JP24K03012).

References

1. Çatal, O., Verbelen, T., Van de Maele, T., Dhoedt, B., Safron, A.: Robot navigation as hierarchical active inference. Neural Networks **142**, 192–204 (2021)
2. Çatal, O., Verbelen, T., Nauta, J., De Boom, C., Dhoedt, B.: Learning perception and planning with deep active inference. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 3952–3956. IEEE (2020)
3. Deng, F., Jang, I., Ahn, S.: Dreamerpro: Reconstruction-free model-based reinforcement learning with prototypical representations, 2021. URL https://arxiv.org/abs/2110.14565
4. Forestier, S., Portelas, R., Mollard, Y., Oudeyer, P.Y.: Intrinsically motivated goal exploration processes with automatic curriculum learning. The Journal of Machine Learning Research **23**(1), 6818–6858 (2022)
5. Fountas, Z., Sajid, N., Mediano, P., Friston, K.: Deep active inference agents using monte-carlo methods. Advances in neural information processing systems **33**, 11662–11675 (2020)
6. Friston, K.: The free-energy principle: a unified brain theory? Nature reviews neuroscience **11**(2), 127–138 (2010)
7. Friston, K., FitzGerald, T., Rigoli, F., Schwartenbeck, P., Pezzulo, G.: Active inference: A process theory. Neural Computation **29**, 1–49 (1 2017)
8. Friston, K., Rigoli, F., Ognibene, D., Mathys, C., Fitzgerald, T., Pezzulo, G.: Active inference and epistemic value. Cognitive neuroscience **6**(4), 187–214 (2015)
9. Ha, D., Schmidhuber, J.: World models. arXiv preprint arXiv:1803.10122 (2018)
10. Hafner, D., Lillicrap, T.P., Fischer, I., Villegas, R., Ha, D., Lee, H., Davidson, J.: Learning latent dynamics for planning from pixels. CoRR **abs/1811.04551** (2018), http://arxiv.org/abs/1811.04551
11. Hafner, D., Lillicrap, T.P., Norouzi, M., Ba, J.: Mastering atari with discrete world models. CoRR **abs/2010.02193** (2020), https://arxiv.org/abs/2010.02193
12. van der Himst, O., Lanillos, P.: Deep active inference for partially observable mdps. In: Active Inference: First International Workshop, IWAI 2020, Co-located with ECML/PKDD 2020, Ghent, Belgium, September 14, 2020, Proceedings 1. pp. 61–71. Springer (2020)
13. Kauvar, I., Doyle, C., Zhou, L., Haber, N.: Curious replay for model-based adaptation. arXiv preprint arXiv:2306.15934 (2023)
14. Kawahara, D., Ozeki, S., Mizuuchi, I.: A curiosity algorithm for robots based on the free energy principle. In: 2022 IEEE/SICE International Symposium on System Integration (SII). pp. 53–59. IEEE (2022)
15. Kingma, D.P., Welling, M., et al.: An introduction to variational autoencoders. Foundations and Trends® in Machine Learning **12**(4), 307–392 (2019)
16. Matsumoto, T., Ohata, W., Benureau, F.C., Tani, J.: Goal-directed planning and goal understanding by extended active inference: Evaluation through simulated and physical robot experiments. Entropy **24** (4 2022). https://doi.org/10.3390/e24040469

17. Mazzaglia, P., Catal, O., Verbelen, T., Dhoedt, B.: Curiosity-driven exploration via latent bayesian surprise. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 7752–7760 (2022)
18. Mazzaglia, P., Verbelen, T., Çatal, O., Dhoedt, B.: The free energy principle for perception and action: A deep learning perspective. Entropy **24**(2), 301 (2022)
19. Mazzaglia, P., Verbelen, T., Dhoedt, B.: Contrastive active inference. In: Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (eds.) Advances in Neural Information Processing Systems. vol. 34, pp. 13870–13882. Curran Associates, Inc. (2021), https://proceedings.neurips.cc/paper/2021/file/73c730319cf839f143bf40954448ce39-Paper.pdf
20. Mendonca, R., Rybkin, O., Daniilidis, K., Hafner, D., Pathak, D.: Discovering and achieving goals via world models. Advances in Neural Information Processing Systems **34**, 24379–24391 (2021)
21. Micheli, V., Alonso, E., Fleuret, F.: Transformers are sample efficient world models. arXiv preprint arXiv:2209.00588 (2022)
22. Pathak, D., Agrawal, P., Efros, A.A., Darrell, T.: Curiosity-driven exploration by self-supervised prediction. In: International conference on machine learning. pp. 2778–2787. PMLR (2017)
23. Pathak, D., Gandhi, D., Gupta, A.: Self-supervised exploration via disagreement. In: International conference on machine learning. pp. 5062–5071. PMLR (2019)
24. Rajeswar, S., Mazzaglia, P., Verbelen, T., Piché, A., Dhoedt, B., Courville, A., Lacoste, A.: Mastering the unsupervised reinforcement learning benchmark from pixels. In: International Conference on Machine Learning. pp. 28598–28617. PMLR (2023)
25. Sancaktar, C., Piater, J., Martius, G.: Regularity as intrinsic reward for free play. arXiv preprint arXiv:2312.01473 (2023)
26. Sancaktar, C., Van Gerven, M.A., Lanillos, P.: End-to-end pixel-based deep active inference for body perception and action. In: 2020 Joint IEEE 10th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob). pp. 1–8. IEEE (2020)
27. Schneider, T., Belousov, B., Abdulsamad, H., Peters, J.: Active inference for robotic manipulation. arXiv preprint arXiv:2206.10313 (2022)
28. Sekar, R., Rybkin, O., Daniilidis, K., Abbeel, P., Hafner, D., Pathak, D.: Planning to explore via self-supervised world models. In: International Conference on Machine Learning. pp. 8583–8592. PMLR (2020)
29. Taniguchi, T., Murata, S., Suzuki, M., Ognibene, D., Lanillos, P., Ugur, E., Jamone, L., Nakamura, T., Ciria, A., Lara, B., et al.: World models and predictive coding for cognitive and developmental robotics: frontiers and challenges. Advanced Robotics **37**(13), 780–806 (2023)
30. Tinguy, D.d., Mazzaglia, P., Verbelen, T., Dhoedt, B.: Home run: Finding your way home by imagining trajectories. In: International Workshop on Active Inference. pp. 210–221. Springer (2022)
31. Ueltzhöffer, K.: Deep active inference. Biological cybernetics **112**(6), 547–573 (2018)

Towards Interaction Design with Active Inference: A Case Study on Noisy Ordinal Selection

Sebastian Stein, John H. Williamson, and Roderick Murray-Smith

School of Computing Science, University of Glasgow, Glasgow, Scotland, UK

Abstract. This paper explores active inference for user interfaces. We implement an active inference approach for 1-of-N selection, a fundamental building block of interactive systems. In this setup, users provide noisy discrete inputs and the interface sequentially identifies an intended target. This problem has an optimal solution (Horstein's algorithm) where the channel noise is iid and known a priori, but is an open problem where the noise is unknown or varying. We reformulate the problem as free energy minimisation and derive a practical active inference implementation. Active inference with a flat noise prior performs comparably to Horstein with conservative noise assumption in the first interaction sequence and as well as Horstein with perfectly calibrated noise thereafter, demonstrating fast adaptation. We also show that active inference can infer the input polarity, offering an extra degree of freedom to users, and adapt to non-stationary noise. The application of active inference to interaction is novel, and we hope this example establishes the groundwork for the community to explore active inference in human-computer interaction.

Code available at https://github.com/drsstein/iwai2024

Keywords: Active Inference · Computational Interaction Design · Ordinal Selection · Adaptive Interfaces · Human Computer Interaction.

1 Introduction

Human computer interaction design is increasingly informed and powered by computational methods: **computational interaction** [12]. Computational interaction optimises interfaces with respect to forward models of user behavior (perception, cognition, motor control, sensor characteristics). This often involves optimisation [2], simulation [11,6], Bayesian inference [17] or reinforcement learning [3,16]. Active inference combines many of the appealing aspects of Bayesian models in interaction design with reasoning over actions that are conventionally approached via RL, but active inference has yet to be applied in a human-computer interaction design context.

Active inference is a model-based control method that minimises expected free energy, simultaneously reasoning about the latent state of its environment and acting to shift the environment state towards some preferred distribution.

Bestowing interfaces with active inference enables them to reason about and adapt to their users and the environment within which they are embedded, while the interaction is ongoing.

Active inference is typically applied to model the behaviour of an agent like a biological system. Here, we characterise the interface as an agent acting under active inference principles whose goal is to extract intention from a user in the face of an unknown and corrupting environment. The agent's goal is to minimise the entropy over the intentional states of the user; it prefers actions that maximise the flow of information from user to system. This is somewhat unfamiliar as an active inference formulation and care is needed in terminology: the agent's *pragmatic goal* is to extract information about user intention and forward it on, but it must also *gain information* about latent states of the user and environment configuration to do so efficiently; the agent wishes to minimise its future surprise about the user's intent. The agent can perceive user actions through sensing, and can act upon the user through the display. This formulation (Figure 1) is a general and powerful way to cast interaction problems as active inference problems.

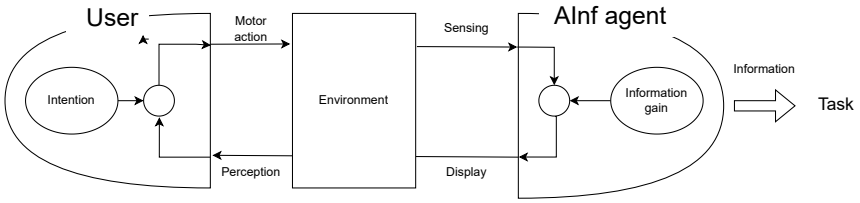


Fig. 1: An active inference agent whose goal is to extract intention from a user's mind and pass it to some external task. The agent has sensing to determine environment states (some of which are influenced by the user) and display channels to influence the environment (some proportion of which are perceived by the user). The agent acts to reduce its uncertainty over user intention so that it can propagate this to the task.

In this paper, we explore the potential for active inference in computational interaction design for noisy ordinal target selection. This is essentially a game where one player (the user) has a number in their mind, and the other player (the computer) tries to work out which number is being thought of by choosing a candidate number and asking the user if the number is higher or lower – with the twist that sometimes the first player lies (noise). Many common interactions can be reduced to 1-of-N selection problems of this nature, either using natural ordering (e.g. alphabets) or imposing an arbitrary order (e.g. a drop-down menu).

2 Related Work

Bayesian models have been explored in the human-computer interaction literature, including at design time (via Bayesian optimisation), interaction time (e.g. via expected information gain) and at evaluation time (Bayesian statistical analyses). Williamson et al. [19] reviews Bayesian models in human-computer interaction generally. With regard to Bayesian approaches to interaction time selection problems, Dasher [18] solved the *entropy-coding problem* of communicating a specific text sequence with a minimum number of inputs, using a feedback model applying arithmetic coding of a statistical language model embedded in a zooming-based interface. [7] used the idea of Bayesian information gain to build a map navigation interface that again minimised the number of inputs required to localise a spatial target. BIGNav formulated the interaction problem as an agent running optimal experiment upon a user. Both Dasher and BIGNav assumed negligible noise in the user inputs.

The *channel coding* problem was explored by Williamson et al. in [20], who proposed the binary selection model explored in this paper and demonstrated robust interface designs for brain-computer interfaces that incorporate Horstein's posterior matching scheme [5]. This results in very robust interfaces in noisy contexts, but is only fully effective for input with known, stationary, independent and identically distributed Bernoulli noise. Simulation as a tool for exploring, optimising and analysing human-computer interaction designs is surveyed in [11], who identify the formulation of explicit *generative* computational models of user (and system) behaviour as a critical step in advancing HCI research. In the active inference literature, [15] explores trust in human–robot collaboration, cast as "mutual predictability" in an active inference framework. [8] describe an active inference approach to an adaptive P300-based text entry BCI, which greatly improved bitrates in communication over a very low capacity speller interfaces. [1] provides a brief outline of the potential use of active inference in brain computer interfaces with the active inference agent engaged in minimising surprise over user intention, following the same line of argument as in this paper. Grizou [4] discuss machine learning approaches to *self-calibrating* interfaces, where the labelling of controls can be arbitrarily permuted but control can still be established; as we will see, self-calibration can arise naturally from an active inference formulation.

3 Problem

The general problem we are interested in is the reliable 1-from-N selection problem (Figure 2), using a binary input subject to random corruption. For example, to pick a number, select a letter, or identify a menu item by pressing one of two buttons. Accumulating several binary button presses is typically required to resolve a particular item. We are interested in *reliable* selection, where users can select items with *arbitrarily low probability of mis-selection* at the cost of an increasing number of binary inputs to identify the target item.

In this abstract setting we ignore the wall clock time to produce each binary decision. This includes the time for the user to reason about the correct decision, the time to actuate the decision and the time to process the feedback from the system about the new state.

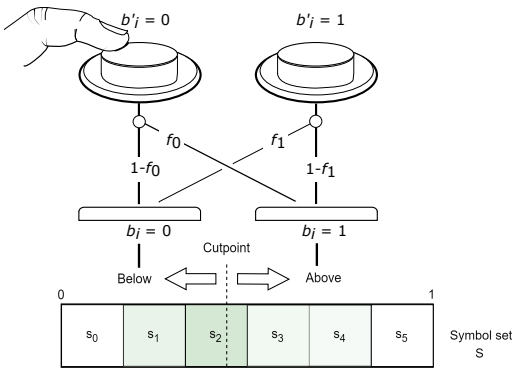


Fig. 2: The ordinal selection problem with noisy binary inputs (adapted from [20]). The user can emit one of two symbols, each with its own probability of flipping.Ordered target symbols are arranged on an interval. A cutpoint m_i is presented to elicit a binary "above/below" decision. A probability distribution over targets (green shading) is maintained during selection.

Although this is an abstract interface model amenable to mathematical modelling, such limited interfaces are practically relevant in brain-computer interfaces such motor imagery EEG control [13] and in other assistive technology input devices such as electromyography where the sensing interface emits binary symbols with very high error levels but where high-capacity, reliable displays are available. The feedback channel is essential; although feedforward error correction (FEC) is commonplace in digital communication, FEC codes are wholly impractical for human input. Feedback error correction is asymptotically more efficient but critically also tractable for human users. We focus on the binary problem for convenience of analysis, but the ideas generalise to any q-ary discrete input device.

3.1 Ordinal selection with noise

Formally, we model an ordinal selection problem, in which the user's task is to identify a discrete symbol $s \in S$ from an ordered sequence of symbols S where typically $|S| = 2^k$. We the user emits a sequence $b_0, b_1, \ldots b_t$ of binary symbols $b_i \in \{0, 1\}$. Timing is ignored and users have a forced choice of binary symbol at each step. There is no explicit selection input like a "click"; decoding is performed once the intended symbol is sufficiently well identified. Each symbol

has an associated random flip probability f_0, f_1, f_0 giving the probability that b_i is flipped from 0 to 1 and vice versa for f_1. We assume iid Bernoulli noise. f_0, f_1 fully characterise the communication channel and are referred to as the **channel statistics**. We can model both symmetric iid noise ($f_0 = f_1$) and asymmetric (biased) noise – particularly relevant in brain-computer interfaces where one input symbol is often significantly more corrupted than the other. Figure 2 illustrates our model.

3.2 Known channel statistics: Horstein decoding

In the case where f_0 and f_1 are known in advance *and* there is an effectively noise-free feedback channel *and* we assume that the symbol set is arbitrarily large, $|S| \to \infty$, then Horstein's posterior matching feedback algorithm [5] is known to be optimal. This algorithm operates on an interval of the real number line $0 \leq x \leq 1$ and forms a probability density $f_X(x)$ over possible values of x. The density $f_X(x)$ is represented as a piecewise linear CDF, and the algorithm progresses by adaptively proposing a series of cutpoints $m_0, m_1, ..., m_t$ based on the history of inputs. Each input rescales the CDF about the proposed cutpoint, giving a closed-form update for the posterior density at each step. Each step of the algorithm is equivalent to a Bayesian update, taking the previous PDF as the distribution over the interval and then updating based on the observed evidence. The cutpoints m_i are simply selected at the median density, which is trivial to evaluate from the piecewise linear CDF.

To convert this to an ordinal selection problem, the interval is subdivided into subintervals $[s_l, s_h]$ each corresponding to an element of S such that the unit interval is completely covered by non-overlapping subintervals. The widths of each element of interval are set to the prior probability of each symbol s_i (with uniform width of size $1/|S|$ if a uniform prior over symbols is assumed). The algorithm terminates either when the density concentrates sufficiently in an interval corresponding to a particular symbol $\int_{s_l}^{s_h} f_X(x) > p_k$ with a simple fixed threshold p_k, or alternatively when the (differential) entropy $H(X)$ decreases by some threshold h_k. To allow for a small degree of mismatch between the true channel statistics and the estimated statistics, a *headroom* f_h is added to f_0, f_1 to product f_0', f_1'; typically f_h is in the range 0.01-0.05. Although the Horstein algorithm is optimal over the continuous space, discretising into a symbol sequence S means that capacity is reduced below the Shannon limit. [20] describes the details of the algorithm and the setting of thresholds.

The known channel statistics problem is solved by Horstein's algorithm modulo details of a particular interface design. However, if f_0, f_1 are *unknown* or changing, or the noise is not iid, the optimal interface for reliable ordinal selection is an open question. Horstein's algorithm is also only optimal for large $|S|$, but many selection problems are from small symbol sets.

4 Method

We assume a model where we have (potentially time varying) probability distributions over the channel statistics $P_t(F_0), P_t(F_1)$ defined by density functions $f_{F_{0,t}}(x)$ and $f_{F_{1,t}}(x)$. Our goal is still to estimate the specific symbol s_i that represents a user's intention. In the unknown channel statistics case, however, we have to trade-off acquiring information about the channel statistics (*channel probes*) and resolving the target symbol. A naïve approach might interleave symbol selection with calibration phases selecting dummy (system-chosen) targets as channel probes. This is simple but inefficient.

We instead choose to formulate this as an active inference problem. We model an agent (AInf agent) whose goal is to perform optimal mind-reading via an unknown channel – to identify the s_i that represents the user's intention with arbitrarily low error rate and in the minimum number of input binary symbols; this implicitly requires online estimation of f_0, f_1. AInf agent must trade off exploration to estimate channel statistics and refine its model of the environment against exploiting the information from the user to identify the hidden target.

Our model formulates the problem as inferring a belief distribution $Q_i(x_i)$ over the unobserved states $x_i : [s^i, f_0^i, f_1^i]$, the unknown symbol s and channel statistics f_0, f_1 at timestep i. The agent's observation at each time step i is the binary symbol $o_i = b_i \in \{0, 1\}$. The agent's only action is to select a cutpoint $m_i \in \mathbb{R}, 0 \leq m_i \leq 1$. The unit interval is mapped onto the ordinal symbols S as described above, using a uniform division of the interval into symbol subintervals, and the symbol is presented to the user. In each step i the agent acquires exactly one binary observation o_i and produces one action m_i.

The belief is initialised with a prior $Q_0(x_0)$ and updated using Bayesian inference. A probabilistic forward model of the environment dynamics $P(x_i|x_{i-1}, m_{i-1})$ is used to propagate the belief through time after every agent action m_i as in Equation (1). A probabilistic forward model of the sensor states $P(b_i|x_i, m_{i-1})$ is used to revise this belief after a new user input is observed as in Equation (2).

$$Q_{i-1}(x_i) = \int P(x_i \,|\, x_{i-1}, m_{i-1})\, Q_{i-1}(x_{i-1})\, \mathrm{d}x_{i-1} \tag{1}$$

$$Q_i(x_i) = \frac{P(b_i \,|\, x_i, m_{i-1})\, Q_{i-1}(x_i)}{P(b_i \,|\, m_{i-1})} \tag{2}$$

$$P(b_i \,|\, m_{i-1}) = \int P(b_i \,|\, x_i, m_{i-1})\, Q_{i-1}(x_i)\, \mathrm{d}x_i \tag{3}$$

AInf agent selects actions m_i by minimising the expected free energy $G(\pi)$ over policies $\pi : (m_i, \ldots, m_{i+T-1})$ with time horizon T and choosing the first action of the optimal policy (4).

$$G(\pi) = \frac{1}{T} \sum_{k=i}^{i+T-1} \underbrace{-\mathbb{E}_{Q_k}\left[D_{\mathrm{KL}}\left(Q_k(x_k) \| Q_{k-1}(x_k)\right)\right]}_{\text{Information gain}} \underbrace{-\mathbb{E}_{Q_k}\left[\ln P_k^c(x_k)\right]}_{\text{Pragmatic value}}, \tag{4}$$

where D_{KL} is the KL-divergence and P^c represent the agent's preferences. In this scenario, the agents goal of identifying the the user's intent is aligned with maximising information gain over the full belief state; in a more complex scenario P^c could encourage specific user behavior akin to *nudging*.

5 Implementation

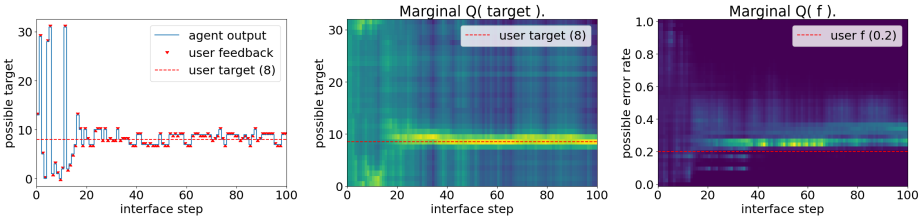


Fig. 3: Interaction between a simulated user with symmetric input error ($f_0 = f_1 = 0.2$) and AInf agent with flat symmetric prior assuming no knowledge about input polarity $Q_0(f_0) = \mathcal{U}(0,1), Q(f_1) = Q(f_0)$. Left: Sequence of agent actions (blue), user input (red arrows), and user target (red dashed). Center: Marginal belief over user targets. Right: Marginal belief over channel statistics. Note that the belief over user target and channel statistics is in the continuous domain and is only discretized for visualisation purposes.

Commonly, the belief Q is approximated using a Gaussian distribution, which leads to poor approximations of the posterior in this problem. A sequence of user inputs is usually consistent with distant mutually exclusive beliefs. For example, successive "below" inputs are consistent with a low target and low error rate, and simultaneously consistent with a high target and high error rate. To model multi-modal distributions, we use a particle filter. A set of $n_p = 100 \cdot |S|$ particles $\{(x_j, w_j)\}$ represent the belief as a mixture of Dirac delta distributions (see Equation (5)), where $w_j \in \mathbb{R}^+$ represent normalised weights $\sum_j w_j = 1$. Particle weights are updated using (2) after every observation, and particles are resampled when the effective sample size $1/\sum w_j^2$ falls below a threshold $\tau_w = 0.5$ using low variance re-sampling. The temporal dynamics of the system are assumed to be stationary with some diffusion $\sigma_d = 0.001$ on the error rate to accommodate drift, updating x_j by sampling from (6).

$$Q(x) = \sum_j w_j \, \delta(x_j) \tag{5}$$

$$P(x_{i+1} \,|\, x_i, m_i) = \mathcal{N}(x_i, \sigma_d^2) \tag{6}$$

Here, the set of possible actions has finite size $|S|$. We exhaustively evaluate all $|S|^T$ policies with time horizon $T = 1$. Where unobserved states are independent of the agent's actions, control reduces to solving a Bandit problem. There,

longer time horizons do not improve action selection. The D_{KL}-divergence in (4) is computed on the Bernoulli distributions w_i and w_{i-1}. Figure 3 shows an example interaction between a simulated user and the agent.

6 Evaluation

We evaluate the performance of the active inference agent through computational simulation experiments designed to answer the following research questions:

- RQ1: How quickly can AINF AGENT infer channel statistics and user targets?
- RQ2: How do AINF AGENT decisions compare to Horstein's algorithm?
- RQ3: How well does AINF AGENT adapt to non-stationary channel statistics?

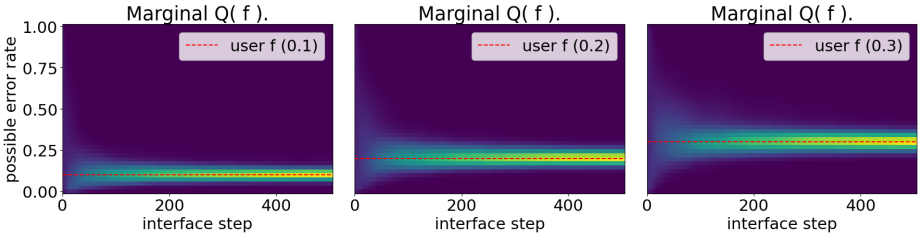


Fig. 4: Marginal probability of the true channel statistics under the belief distribution. Mean marginal belief distribution across all 32 user targets and 10 repetitions each over five successive interaction episodes of 100 steps. The belief converges around the true channel statistics within 100 to 300 steps, with higher error rates taking longer to infer.

RQ1: How quickly can AINF AGENT infer channel statistics and user targets? We simulated users with symmetric channel statistics $f_0 = f_1 \in [0.1, 0.2, 0.3]$ and every symbol as target $0 \leq s < |S| = 32$, each interacting with AINF AGENT 10 times for 500 steps resulting in $3 \times 32 \times 10 = 960$ simulation runs. AINF AGENT belief was initialised assuming error symmetry $f_0 = f_1$ and *without* assuming knowledge about input polarity $Q_0(f) = \mathcal{U}(0, 1)$. Within each run, the agent's belief over user targets was reset to a flat prior every 100 steps, while the belief over channel statistics was maintained, simulating five sequential episodic interactions. We measured the probability of true error $f^* \pm \epsilon$ under the belief distribution at every step in each run, shown in aggregate in Figure 4. Starting from a flat prior, the belief about the true channel statistics increases over time demonstrating that they are being identified. Smaller error rates are identified faster than higher error rates. Under RQ3 below we show that AINF AGENT target selection behavior is comparable to that under perfect knowledge of channel statistics after 1-3 episodes.

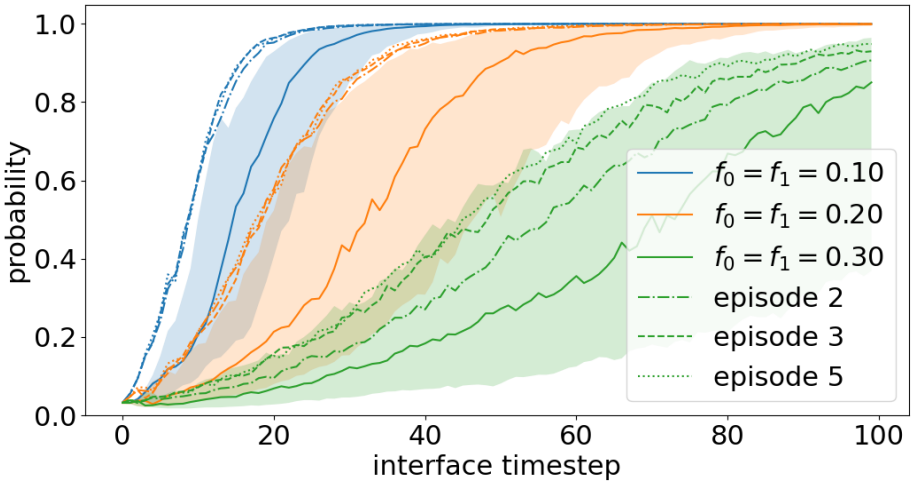


Fig. 5: Marginal probability of the true user target under the belief distribution. Shown are the median and inter-quartile range across all 32 user targets and 10 repetitions in the first 100 interaction steps, and the median across successive interaction episodes. Target inference speed converges after 1-3 episodes demonstrating the saturating effect of inferring the channel statistics.

We evaluate user target inference on the same set of simulations, measuring the probability of the user's target under the belief distribution in the first and subsequent episodes (Figure 5). User targets are inferred faster when error rates are lower, and the number of steps required to infer the user target converges after 1-2 episodes, demonstrating the saturating benefit of inferring channel statistics on target inference.

RQ2: How does AINF AGENT *decision-making compare to the Horstein's algorithm?* We evaluated decision-making performance using speed-accuracy trade-off curves. We measured the decision accuracy (higher is better) and the number of interaction steps until a decision was taken (lower is better) under a decision rule that selects the most likely symbol $\operatorname{argmax} Q(s)$ under the belief distribution when $\max Q(s) \geq \tau$ for varying thresholds $\tau \in [0, 1]$.

AINF AGENT decision-making performance was evaluated on the same simulations as in RQ1 above. We repeated these simulations with different priors over channel statistics $Q_0(f) = \mathcal{U}(0, 0.5)$. We shall refer to this set of simulations as *known control polarity* and to the original set of simulations as *unknown control polarity*, as error rates greater than 0.5 can be interpreted as predominantly, possibly intentionally, sending the flipped signal. This problem can arises in several interaction contexts. For example, joystick motion "away from" and "towards" the user's body is mapped differently onto "up" and "down" control actions in aeroplane and helicopter control. Inferring and adapting to polarity in the user's mental model of motion-to-control mappings offers an extra degree of freedom to

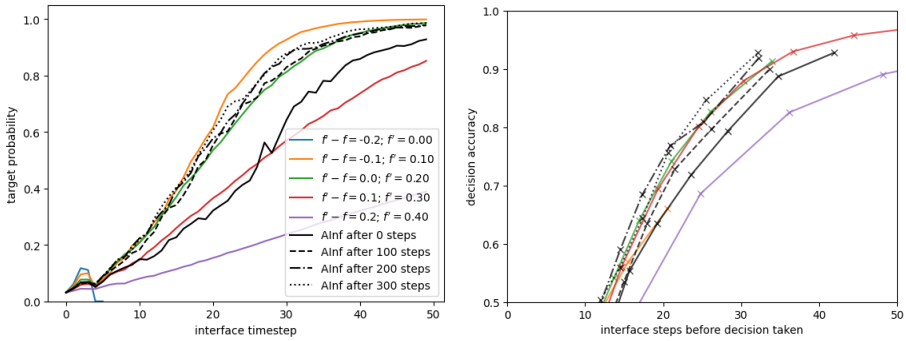


Fig. 6: Decision-making performance of AInf agent and Horstein's algorithm with *known control polarity*. Median marginal belief probability of the true user target for $f_0 = f_1 = 0.2$, varied assumed headroom $f' - f$ for the Horstein algorithm, and sequential episodes for the AInf agent.

users [14]. To evaluate decision-making performance of Horstein's algorithm, we simulated interactions of users with symmetric channel statistics $f_0 = f_1 = 0.2$ and every symbol as target $0 \leq s < |S| = 32$, each interacting with a variant of Horstein's algorithm 100 times for 100 steps. Ten variants of Horstein's algorithm with different *headroom* $(f' - f) \in [-0.2, -0.1, 0, 0.1, \ldots 0.8]$ were used, resulting in $32 \times 100 \times 10 = 32000$ simulation runs. In every run, the belief over user targets was initialised to a flat prior.

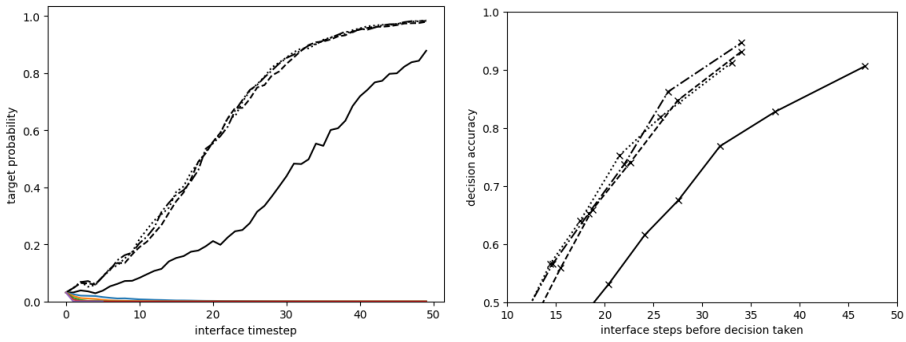


Fig. 7: Decision-making performance of AInf agent and Horstein's algorithm with *unknown control polarity*. Median marginal belief probability of the true user target for $f_0 = f_1 = 0.2$, varied assumed headroom $f' - f$ for the Horstein algorithm, and sequential episodes for the AInf agent.

The target probability over time and speed-accuracy curves are shown for the *known control polarity* and *unknown control polarity* conditions in Figures 6 and

7, respectively. In the *known control polarity* condition, AInf agent target inference speed and decision-making performance are comparable to Horstein's algorithm with perfect knowledge of the channel statistics after 200 steps, and outperforms it slightly thereafter. This demonstrates that inferring channel statistics online from a flat prior is feasible, and that AInf agent can address the reliable ordinal selection. Horstein's algorithm decision-making performance appeared very sensitive to its *headroom*. Horstein's algorithm with negative headroom, underestimating error rates, and headroom above 50%, i.e. mismatched polarity, failing catastrophically (speed-accuracy curves are near zero accuracy across all timesteps. Where the headroom was too conservative ($f' - f = 0.2$), overestimating error rates, accuracy was around 10% lower at the same number of interaction steps compared to the best performing Horstein variant.

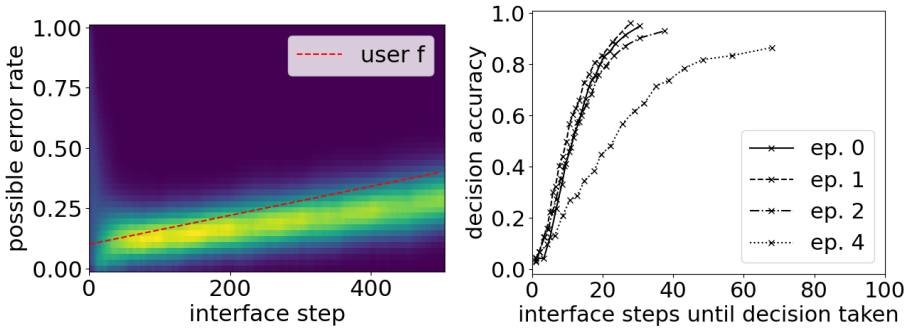


Fig. 8: Marginal belief distribution over channel statistics (left) and decision-making performance (right) of AInf agent with *unknown control polarity* and non-stationary channel statistics.

RQ3: How well does AInf agent adapt to non-stationary channel statistics? Above, we have shown that AInf agent can infer and adapt to unknown channel statistics assuming those statistics are stationary. To explore how well AInf agent responds to smooth changes in the channel statistics, we performed a series of simulations as in the *unknown control polarity* condition, but with symmetric error rate increasing linearly from 0.1 to 0.3 over 5 successive episodes of 100 timesteps. We increased the diffusion scale to $\sigma_d = 0.01$ to make such changes more likely under the agent's model. We evaluated how well AInf agent can track changing channel statistics by estimating the mean marginal belief distribution over error rates across timesteps, and the decision-making performance via the speed-accuracy curve (see Figure 8). While the agent is more uncertain about the channel statistics overall, it continues to provide a good speed-accuracy trade-off throughout the period of error rate degradation.

In preliminary experiments not shown here for lack of space, we found AInf agent decision-making performance to remain stable under a wide range of par-

ticle filter hyperparameter settings, including changes to the number of particles n_p, resampling threshold τ_w and amount diffusion scale σ_d.

7 Discussion

We presented an active inference approach to reliable selection with two noisy inputs. This is human-computer interaction stripped back to its barest elements, but still complex enough to represent real interactive systems. We formulate the interface as an independent agent charged with facilitating the flow of information, acting as an active transducer able to reason about the environment and user characteristics to optimise this flow. Active inference gives a high-level formulation of the problem that is fully Bayesian and flexible enough to precisely model this task. Classical information theoretic approaches are only optimal under assumptions that are rarely met in the messiness of human interactions.

We focused on the control polarity and non-stationary channel statistics, but it would be quite feasible to relax other assumptions in the active inference formulation. These include the iid noise assumption (for example, modelling bursty or otherwise correlated noise), time-varying numbers of input symbols or input symbols with different costs. This scenario often arises in assistive technology [10] where some inputs may have high physical demands or long refractory periods.

Approximate Bayesian inference using a particle filter is well suited to modelling multi-modal distributions but has a high computational cost. As we move from the most elementary interactive systems explored in this paper, we will need to evaluate these more complex tasks with real users. To achieve real-time (sub-second) closed-loop performance with an active inference approach we may need to implement amortized inference approaches, such as proposed in [9].

In the ordinal selection task, stationarity in the unobserved state and alignment of the agent's goal with maximising information gain simplified the inference problem, but time-varying dynamics and exploration-exploitation trade-offs abound in human-computer interaction tasks. Relying only on forward models of users and applications, active inference can help make offline and online interaction design more transparent and modular. The unifying approach of active inference holds real promise in human-computer interaction, bringing together Bayesian models of interaction with explicit reasoning over future actions. While computational demands and challenging modelling work lie ahead, as we have demonstrated in this paper, even simple and well-understood interaction problems can be turbocharged by an active inference perspective.

Acknowledgements

This research was supported by the project Designing Interaction Freedom via Active Inference (DIFAI). The project was selected by the ERC (Advanced Grant proposal 101097708) and was funded by the UK Horizon guarantee scheme as Engineering and Physical Sciences Research Council (EPSRC) project EP/Y029178/1. We thank Aegean Airlines S.A. for their donation.

References

1. Ather, S.H.: Active inference as a framework for brain-computer interfaces. bioRxiv pp. 2021–02 (2021)
2. Dayama, N.R., Todi, K., Saarelainen, T., Oulasvirta, A.: GRIDS: Interactive Layout Design with Integer Programming. In: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. pp. 1–13. ACM, Honolulu HI USA (Apr 2020). https://doi.org/10.1145/3313831.3376553
3. Glowacka, D., Howes, A., Jokinen, J.P., Oulasvirta, A., Şimşek, Ö.: RL4HCI: Reinforcement Learning for Humans, Computers, and Interaction. In: Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems. pp. 1–3. ACM, Yokohama Japan (May 2021). https://doi.org/10.1145/3411763.3441323
4. Grizou, J.: Interactive introduction to self-calibrating interfaces (Dec 2022). https://doi.org/10.48550/arXiv.2212.05766
5. Horstein, M.: Sequential transmission using noiseless feedback. IEEE Transactions on Information Theory **9**(3), 136–143 (1963)
6. Ikkala, A., Fischer, F., Klar, M., Bachinski, M., Fleig, A., Howes, A., Hämäläinen, P., Müller, J., Murray-Smith, R., Oulasvirta, A.: Breathing Life Into Biomechanical User Models. In: Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology. pp. 1–14. ACM, Bend OR USA (Oct 2022). https://doi.org/10.1145/3526113.3545689
7. Liu, W., D'Oliveira, R.L., Beaudouin-Lafon, M., Rioul, O.: BIGnav: Bayesian Information Gain for Guiding Multiscale Navigation. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. pp. 5869–5880. ACM, Denver Colorado USA (May 2017). https://doi.org/10.1145/3025453.3025524
8. Mladenovic, J., Frey, J., Joffily, M., Maby, E., Lotte, F., Mattout, J.: Active Inference as a unifying, generic and adaptive framework for a P300-based BCI. Journal of Neural Engineering **17**(1), 016054 (2020)
9. Moon, H.S., Liao, Y.C., Li, C., Lee, B., Oulasvirta, A.: Real-time 3d target inference via biomechanical simulation. In: Proceedings of the CHI Conference on Human Factors in Computing Systems. pp. 1–18 (2024)
10. Müller-Putz, G.R., Ofner, P., Schwarz, A., Pereira, J., Luzhnica, G., di Sciascio, C., Veas, E., Stein, S., Williamson, J., Murray-Smith, R., et al.: Moregrasp: Restoration of upper limb function in individuals with high spinal cord injury by multimodal neuroprostheses for interaction in daily activities (2017)
11. Murray-Smith, R., Oulasvirta, A., Howes, A., Müller, J., Ikkala, A., Bachinski, M., Fleig, A., Fischer, F., Klar, M.: What simulation can do for hci research. Interactions **29**(6), 48–53 (2022)
12. Oulasvirta, A., Kristensson, P.O., Bi, X., Howes, A.: Computational Interaction. Oxford University Press (2018)
13. Padfield, N., Zabalza, J., Zhao, H., Masero, V., Ren, J.: EEG-Based Brain-Computer Interfaces Using Motor-Imagery: Techniques and Challenges. Sensors (Basel, Switzerland) **19**(6), 1423 (Mar 2019). https://doi.org/10.3390/s19061423
14. Reddy, S., Levine, S., Dragan, A.D.: First contact: Unsupervised human-machine co-adaptation via mutual information maximization. arXiv preprint arXiv:2205.12381 (2022)
15. Schoeller, F., Miller, M., Salomon, R., Friston, K.J.: Trust as extended control: Human-machine interactions as active inference. Frontiers in Systems Neuroscience p. 93 (2021)

16. Todi, K., Bailly, G., Leiva, L., Oulasvirta, A.: Adapting user interfaces with model-based reinforcement learning. In: Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. pp. 1–13 (2021)
17. Velloso, E., Morimoto, C.H.: A Probabilistic Interpretation of Motion Correlation Selection Techniques. In: Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. pp. 1–13. ACM, Yokohama Japan (May 2021). https://doi.org/10.1145/3411764.3445184
18. Ward, D.J., Blackwell, A.F., MacKay, D.J.C.: Dasher—a data entry interface using continuous gestures and language models. In: Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology - UIST '00. pp. 129–137. ACM Press, San Diego, California, United States (2000). https://doi.org/10.1145/354401.354427
19. Williamson, J.H.: An Introduction to Bayesian Methods for Interaction Design. Bayesian Methods for Interaction and Design **1** (2022)
20. Williamson, J.H., Quek, M., Popescu, I., Ramsay, A., Murray-Smith, R.: Efficient human-machine control with asymmetric marginal reliability input devices. Plos one **15**(6), e0233603 (2020)

Online Structure Learning with Dirichlet Processes through Message Passing

Bart van Erp[1,2], Wouter W. L. Nuijten[1], and Bert de Vries[1,2,3]

[1] Eindhoven University of Technology, 5612 AP Eindhoven, The Netherlands
[2] Lazy Dynamics, 5611 XD Eindhoven, The Netherlands
[3] GN Hearing, 5612 AB Eindhoven, The Netherlands

Abstract. Generative or probabilistic modeling is crucial for developing intelligent agents that can reason about their environment. However, designing these models manually for complex tasks is often infeasible. Structure learning addresses this challenge by automating model creation based on sensory observations, balancing accuracy with complexity. Central to structure learning is Bayesian model comparison, which provides a principled framework for evaluating models based on their evidence. This paper focuses on model expansion and introduces an online message passing procedure using Dirichlet processes, a prominent prior in non-parametric Bayesian methods. Our approach builds on previous work by automating Bayesian model comparison using message passing based on variational free energy minimization. We derive novel message passing update rules to emulate Dirichlet processes, offering a flexible and scalable method for online structure learning. Our method generalizes to arbitrary models and treats structure learning identically to state estimation and parameter learning. The experimental results validate the effectiveness of our approach on an infinite mixture model.

Keywords: Dirichlet processes · Factor graphs · Infinite mixture model · Message passing · Probabilistic inference · Scale factors · Structure learning.

1 Introduction

The task of generative or probabilistic modeling is fundamental in developing intelligent agents capable of reasoning about their environment. However, it is often infeasible for human engineers to manually design these models for complex tasks because of the involved intricacies. Structure learning addresses this challenge by automating the construction of models based on sensory observations, thus alleviating the burden on human engineers.

Structure learning is encapsulated in the task of Bayesian model comparison, which provides a principled framework for comparing models based on their evidence. This process facilitates the identification of better models that are either smaller or larger than a baseline, known, respectively, as model reduction [4, 14, 15] and model expansion [16, 31]. These techniques are critical for refining

and optimizing models, thus enhancing their performance and applicability in various tasks. This paper will focus on model expansion in particular.

In [9] the tasks of Bayesian model comparison [18], selection and combination [25] have been automated using message passing based on variational free energy minimization. This paper extends this set of methods with Dirichlet processes [6, 10, 26, 33], which are one of the most established priors in non-parametric Bayes. Effectively, we present an online message passing procedure based on Dirichlet processes which enables the model to grow automatically over time, providing a natural trade-off between model accuracy and complexity.

Our approach shows similarities with [36], yet offers more flexibility as a result of our commitment to message passing. Compared to [23], our approach leverages scale factors [27, 29, Ch.6] to track the model evidence rather than performing a partial mean-field approximation. In contrast to [16, 31] our approach is based on non-parametric priors, allowing for a message passing-based treatment of both state estimation, parameter learning and structure adaptation, which is not limited to discrete-space models.

This paper presents a novel and principled approach to online structure learning using message passing. Specifically, we make the following contributions:

- We present a generic and modular approach similar to the sequential updating and greedy search algorithm [36] for online structure learning utilizing Dirichlet processes;
- We derive novel message passing update rules to emulate Dirichlet processes, based on the mixture node recently introduced in [9];
- We demonstrate our approach on an infinite mixture model [28], with the potential for generalization to arbitrary models.

To provide a solid foundation for all readers, Section 2 introduces Forney-style factor graphs and message passing, the core methodology behind this paper. Readers unfamiliar with this methodology and its benefits are encouraged not to skip this section. Throughout the subsequent sections, we use the infinite mixture model [28], as specified in Section 3, as a running example to elucidate our approach. It should be noted, however, that the methods presented in this paper can be easily generalized to more complex graphs due to the inherent modularity of our approach. Using this model, Section 4 details how inference can be executed to ensure the message passing procedure emulates a Dirichlet process, facilitating online structure learning. The experimental results validating our approach are presented in Section 5, and Section 6 follows with a discussion of the presented approach, concluding the paper.

2 Technical background

This section provides a concise review of factor graphs and message passing algorithms, essential for understanding our core contributions. For a deeper understanding, we provide references rather than an exhaustive review. In Section 2.1,

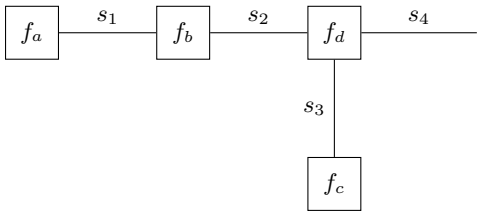


Fig. 1: A Forney-style factor graph representation of the factorization in (2).

we introduce factor graphs for visualizing factorizable probabilistic models, after which Section 2.2 covers efficient probabilistic inference through message passing. Section 2.3 explains how to track model evidence locally with message passing using scale factors.

2.1 Forney-style factor graphs

A factor graph is a type of probabilistic graphical model. We use the Forney-style factor graph (FFG) framework from [11] with notations from [21] to visualize our models. An FFG represents a factorized function:

$$f(s) = \prod_{a \in \mathcal{V}} f_a(s_a), \tag{1}$$

where s includes all variables, and $s_a \subseteq s$ includes the variables of factor f_a. In an FFG, nodes ($\mathcal{V}$) represent factors, and edges ($\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$) represent variables. An edge connects to a node if the variable is an argument of the factor at that node. The edges connected to node $a \in \mathcal{V}$ are denoted by $\mathcal{E}(a)$, and the nodes connected to edge $i \in \mathcal{E}$ are denoted by $\mathcal{V}(i)$. For example, consider

$$f(s_1, s_2, s_3, s_4) = f_a(s_1) f_b(s_1, s_2) f_c(s_3) f_d(s_2, s_3, s_4), \tag{2}$$

which represents a factorization whose FFG representation is shown in Figure 1. For a detailed review of factor graphs, see [21, 22].

2.2 Sum-product message passing

Consider the normalized probabilistic model

$$p(y, s) = \prod_{a \in \mathcal{V}} f_a(y_a, s_a), \tag{3}$$

where y represents observed variables and s represents latent variables. The subset $y_a \subseteq y$ can be empty, such as in prior distributions. Upon observing realizations $\hat{y}$, the model $p(y = \hat{y}, s)$ becomes unnormalized. Probabilistic inference involves computing the posterior distribution $p(s \mid y = \hat{y})$ and the model evidence $p(y = \hat{y})$ as the decomposition $p(y = \hat{y}, s) = p(s \mid y = \hat{y}) p(y = \hat{y})$.

Consider integrating over all variables in the model except s_j as $\int p(y = \hat{y}, s)\,\mathrm{d}s_{\setminus j}$. This integration can be performed through smaller local computations, whose results are termed messages, which propagate over the graph edges. The sum-product message $\vec{\mu}_{s_j}(s_j)$ flowing from node $f_a(y_a = \hat{y}_a, s_a)$ with incoming messages $\vec{\mu}_{s_i}(s_i)$ is given by [19]

$$\vec{\mu}_{s_j}(s_j) = \int f_a(y_a = \hat{y}_a, s_a) \prod_{\substack{i \in \mathcal{E}(a) \\ i \neq j}} \vec{\mu}_{s_i}(s_i)\,\mathrm{d}s_{a\setminus j}. \tag{4}$$

Edges in the graph are represented by directed arrows to distinguish between forward ($\vec{\mu}_{s_j}(s_j)$) and backward ($\overleftarrow{\mu}_{s_j}(s_j)$) messages. For acyclic models, the global integration reduces to the product of messages

$$\int p(y = \hat{y}, s)\,\mathrm{d}s_{\setminus j} = \vec{\mu}_{s_j}(s_j)\overleftarrow{\mu}_{s_j}(s_j). \tag{5}$$

Posterior distributions can be obtained by normalizing the resulting product. The computed normalization constant represents the model evidence. For derivations of the message passing update rules, see [37]. Variations of this approach also yields alternative algorithms such as variational message passing [35], expectation propagation [24], expectation maximization [8], and hybrid algorithms.

2.3 Scale factors

The previously discussed integration $\int p(y = \hat{y}, s)\,\mathrm{d}s_{\setminus j}$ can be expressed as

$$\int p(y = \hat{y}, s)\,\mathrm{d}s_{\setminus j} = p(y = \hat{y}) \int p(s \mid y = \hat{y})\,\mathrm{d}s_{\setminus j} = p(y = \hat{y})p(s_j \mid y = \hat{y}), \tag{6}$$

where $p(s_j \mid y = \hat{y})$ is the marginal distribution of s_j. This means that the product of two colliding sum-product messages $\vec{\mu}_{s_j}(s_j)\overleftarrow{\mu}_{s_j}(s_j)$ in an acyclic graph yields the scaled marginal distribution $p(y = \hat{y})p(s_j \mid y = \hat{y})$. Thus, we can obtain both the normalized posterior $p(s_j \mid y = \hat{y})$ and the model evidence $p(y = \hat{y})$ at any edge or node in the graph.

Theorem 1. *[9, Theorem 1] Consider an acyclic Forney-style factor graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. The model evidence of the corresponding model $p(y = \hat{y}, s)$ can be computed at any edge in the graph as $\int \vec{\mu}_{s_j}(s_j)\overleftarrow{\mu}_{s_j}(s_j)\,\mathrm{d}s_j$ for all $j \in \mathcal{E}$ and at any node in the graph as $\int f_a(y_a = \hat{y}_a, s_a) \prod_{i \in \mathcal{E}(a)} \vec{\mu}_{s_i}(s_i)\,\mathrm{d}s_a$ for all $a \in \mathcal{V}$.*

This local computation of model evidence is enabled by the scaling of messages resulting from the equality in (4). Consequently, the messages $\vec{\mu}_{s_j}(s_j)$ can be decomposed as

$$\vec{\mu}_{s_j}(s_j) = \vec{\beta}_{s_j}\vec{p}_{s_j}(s_j), \tag{7}$$

where $\vec{p}_{s_j}(s_j)$ is the normalized probability distribution of the message $\vec{\mu}_{s_j}(s_j)$, and $\vec{\beta}_{s_j}$ is the scaling factor [29, Ch.6], [27]. These scale factors serve as local summaries of the model evidence passed along the graph.

3 Model specification

In this section, we describe the probabilistic model that underpins our novel inference approach detailed in Section 4. As a running example, we employ the infinite mixture model [28]. This model leverages the unique properties of Dirichlet processes, allowing the model to expand dynamically over time. Consequently, it serves as an ideal and principled example for structure learning.

The infinite mixture model works as follows. Consider a single observation y_n, which is modelled by a likelihood model $p(y_n \mid \theta, c_n)$, with parameters θ. The model assumes multiple possible options or regimes for the parameters depending on the cluster assignment probability c_n. This cluster assignment probability c_n comprises a 1-of-K binary vector with elements $c_{nk} \in \{0, 1\}$ constrained by $\sum_{k=1}^{K} c_{nk} = 1$. Depending on the class, the observation is modelled by a different set of parameters. When the k^{th} class is active, the corresponding set of parameters is given by θ_k. As a result, the likelihood model can be further factorized as

$$p(y_n \mid \theta, c_n) = \prod_{k=1}^{\infty} p(y_n \mid \theta_k)^{c_{nk}}. \tag{8}$$

The infinite mixture model assumes we have an infinite amount of classes ($K = \infty$). Although this might seem computationally intractable, in practice only a limited number of classes is active as we will show in Section 4. The model's strength lies in its ability to grow the number of active classes over time, providing opportunities to expand the model in a principled manner.

In addition to the likelihood model, we define the prior over the cluster parameters as the base distribution G_0 as

$$p(\theta_k) = G_0(\theta_k) \quad \forall\, k. \tag{9}$$

Here independence across the clusters is implied by the characterization of [20] as

$$p(\theta) = \prod_{k=1}^{\infty} p(\theta_k), \tag{10}$$

and will also result into independence across the posteriors over the clusters [34].

The cluster assignment probabilities c_n are modeled using a categorical distribution

$$p(c_n \mid \pi) = \text{Cat}(c_n \mid \pi), \tag{11}$$

with event probabilities π. The prior on the event probabilities is in our case defined as

$$p(\pi) = \lim_{K \to \infty} \text{Dir}\left(\pi \,\middle|\, \frac{\alpha}{K} 1_K\right), \tag{12}$$

with α representing the concentration parameter and 1_K denoting a vector of ones of length K. Alternative definitions are also possible, e.g. using the Griffiths-Engen-McCloskey (GEM) distribution or using the stick-breaking representation, however, for ease of inference in Section 4 we use the former. Together,

the base distribution G_0 and the concentration parameter α characterize the underlying Dirichlet process.

With all the individual elements identified, the full generative model of the infinite mixture model can now be constructed for multiple observations. Given N observations $y = \{y_1, y_2, \ldots, y_N\}$ the total model factorizes as

$$p(y, \theta, c, \pi) = \underbrace{p(\pi)}_{(12)} \prod_{k=1}^{\infty} \underbrace{p(\theta_k)}_{(9)} \prod_{n=1}^{N} \underbrace{p(y_n \,|\, \theta, c_n)}_{(8)} \underbrace{p(c_n \,|\, \pi)}_{(11)}. \tag{13}$$

4 Probabilistic inference

With the model described in the previous section, we will now show how we can perform inference in this model. Specifically, we are interested in computing the marginal posterior distributions $p(\pi \,|\, y_{\leq n})$ and $p(\theta_k \,|\, y_{\leq n}) \,\forall\, k$ in the infinite mixture model. We focus here on online inference, where this inference task is performed using streaming data, as we highlight the importance of in-the-field structure learning. Ideally, we wish to solve the discrete equivalent of the Chapman-Kolmogorov integral [32, Ch.4]

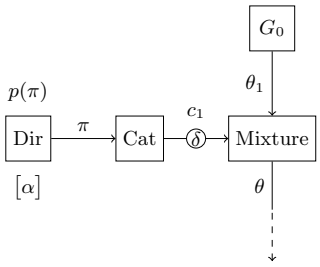
$$p(\theta, \pi \,|\, y_{\leq n}) \propto \sum_{c_n} p(y_n \,|\, \theta, c_n)\, p(c_n \,|\, \pi)\, p(\theta, \pi \,|\, y_{<n}) \tag{14}$$

recursively. However, the infinite dimensionality of c_n results in intractable inference. To circumvent this problem, all inactive components where the posterior beliefs over the parameters have not yet been updated from the prior belief, are grouped together. The components or cluster with indices $k > K^*$, where K^* denotes the number of active components, are grouped into a single component with concentration parameter $\lim_{K\to\infty} \sum_{k=K^*+1}^{K} \alpha/K = \alpha$. This particular grouping is very beneficial as the problem can now be tackled as a standard model comparison task as in [9].

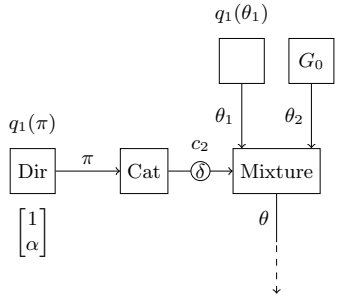
Furthermore, to limit the number of excitable components per observation, the class label c_n is constrained to correspond to a single class, such that each data point can only belong to a single class and therefore has the potential to initiate no more than a single component, as described in [36]. This constraint is reflected by constraining the approximate marginal distribution $q(c_n)$ [37] to

$$q(c_n) = \delta[c_n - e_k], \qquad \text{s.t. } k = \arg\max_k \vec{\mu}_{c_n}(c_n = e_k)\vec{\mu}_{c_n}(c_n = e_k), \tag{15}$$

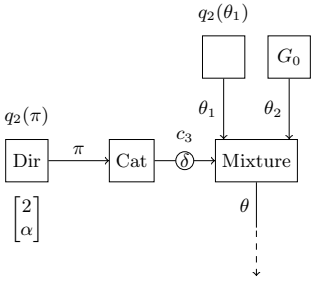
where $\delta[\cdot]$ denotes the Kronecker delta function and where we pick the component c_n as the maximum a posteriori estimate. This is similar to the Bayesian model selection setup as described in [9, Sec.5.2]. Using the Chinese restaurant metaphor, this constraint enforces that every customer can only sit at a single table at once. If we would not enforce this constraint, then there would always be a non-zero probability of the observation originating from the group of inactive components, which would be sufficient to activate a new component for each observation.



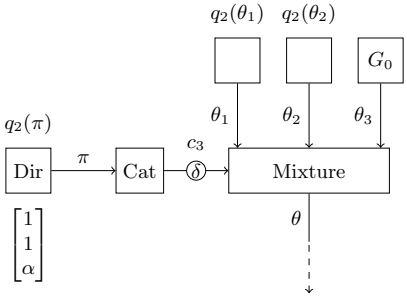
(a) Factor graph visualization for the first observation.



(b) Factor graph visualization for the second observation, where the first observation has initiated a new component.



(c) Factor graph visualization for the third observation, where the first and second observation are assigned to the same component.



(d) Factor graph visualization for the third observation, where the first and second observation have initiated distinct components.

Fig. 2: Factor graphs of the initial time slices of the infinite mixture model of Section 3. The edge denoted by θ here denotes the active or selected parameter settings. The edges connected to θ have been dashed to highlight its extensibility towards arbitrary observation models. The parameter vector below the Dir-node denotes the simplified vector of concentration parameters.

Based on this constraint, we approximate the marginal posterior distributions over $\theta_k \,\forall\, k$ and π with approximate posterior distributions $q_n(\theta_k)$ and $q_n(\pi)$, where the subscript n explicitly indexes the latest observation. Online inference proceeds using the iterative update procedure for θ_k as

$$q_n(\theta_k) \propto \begin{cases} p(y_n \,|\, \theta_k) q_{n-1}(\theta_k), & \text{if } q(c_n) = \delta[c_n - e_k], \\ q_{n-1}(\theta_k), & \text{otherwise,} \end{cases} \tag{16}$$

which effectively states that only the parameters gets updated which have been most likely to have generated the data. The posterior belief over π gets updated

as

$$q_n(\pi) \propto q_{n-1}(\pi) \underbrace{\sum_{c_n} q(c_n) p(c_n \,|\, \pi)}_{\overleftarrow{\mu}_\pi(\pi)} . \tag{17}$$

The initial conditions of this recursion find their origin in the model specification and are specified by

$$q_0(\theta_k) = p(\theta_k), \tag{18a}$$

$$q_0(\pi) = p(\pi). \tag{18b}$$

The above inference procedure can be automated using an adapted version of the mixture node from [9] as presented in Table 1. Effectively this node internally computes the model evidences of the individual combinations of inputs and output using the scale factors from Section 2.3, which are an indicator for how likely a data point y_n originated from one particular set of parameters θ_k. Through normalization of these evidences and together with the prior on the class label c_n, one can obtain the posterior distribution of the class label. In comparison to the mixture node as introduced in [9] the only adaptation occurs in the backward messages towards the parameters. This adaptation entails that only the parameters of the active component are being updated. Figure 2 visualizes the factor graphs corresponding to the initial time slices of the online training procedure. From this figure it can also be seen how the mixture node of [9] be used to represent the infinite mixture model.

The biggest benefit of this approach is that the system is inherently modular. The likelihood and priors can be extended to arbitrarily complex or hierarchical models to model more complex phenomena. By adding a temporal dependency $p(c_n \,|\, c_{n-1})$ to the model, one effectively creates a sticky Dirichlet process [12,13]. With the message passing updates rules from Table 1 together with rules derived in earlier works, e.g. [21, 27], one can build arbitrarily complex graphs tailored to any problem.

5 Experiments

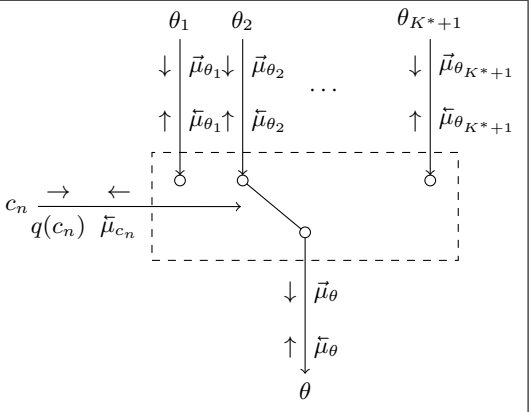
All experiments have been performed using the scientific programming language `Julia` [5] with the state-of-the-art probabilistic programming package `RxInfer.jl` [2]. The mixture node specified in Table 1 has been integrated in its dependency `ReactiveMP.jl` [1,3]. In addition to the results presented in the upcoming subsections, interactive `Pluto.jl` notebooks are made available online[4], allowing the reader to change hyperparameters in real-time.

For online learning of the infinite mixture model as described in Section 4, we generate observations from a two-dimensional normal mixture model with 8 clusters. As a model for these generated observations, we pick the infinite mixture model of (13). Here, the likelihood model is set to $p(y_n \,|\, \theta_k) = \mathcal{N}(y_n \,|\, \theta_k, \mathrm{I}_2)$

[4] All experiments are publicly available at `https://github.com/biaslab/OnlineMessagePassingDirichletProcess`.

Table 1: Table containing (top) the Forney-style factor graph representation of the mixture node of [9]. The edge denoted by θ here denotes the active or selected parameter settings. (bottom) The derived outgoing messages for the mixture node mimicing a Dirichlet process. It can be noted that the backward message towards c_n resembles a scaled categorical distribution and that the forward message towards θ represents only one of the incoming messages $\vec{\mu}_{\theta_k}(\theta)$.

Factor node



Messages	**Functional form**
$\overleftarrow{\mu}_{c_n}(c_n)$	$\prod_{k=1}^{K^*+1} \left(\int \vec{\mu}_{\theta_k}(\theta_k) \overleftarrow{\mu}_{\theta_k}(\theta_k) \mathrm{d}\theta_k \right)^{c_{nk}}$
$\vec{\mu}_\theta(\theta)$	$\vec{\mu}_{\theta_k}(\theta)$ if $q(c_n) = \delta[c_n - e_k]$
$\overleftarrow{\mu}_{\theta_k}(\theta_k)$	$\begin{cases} \overleftarrow{\mu}_\theta(\theta_k) & \text{if } q(c_n) = \delta[c_n - e_k] \\ const & \text{otherwise} \end{cases}$

where I_2 denotes a two-dimensional identity matrix. The priors over the mean parameters θ_k are set to be uninformative as $p(\theta_k) = \mathcal{N}(\theta_k \,|\, 0_2, 10\mathrm{I}_2)$, where 0_2 denotes a two-dimensional vector of zeros. The concentration parameter is initialized as $\alpha = 0.1$.

Figure 3 shows the inferred class assignments and posterior mean parameters, together with the posterior concentration parameters, as a function of the number of observations. From the figure we can validate that the inference procedure indeed recovers the 8 clusters that were used to generate the data. Furthermore, the mean parameters have converged to the data generating cluster means.

6 Discussion and conclusions

The presented approach in this paper enables model expansion by emulating Dirichlet processes through message passing. Through the use of scale factors, we

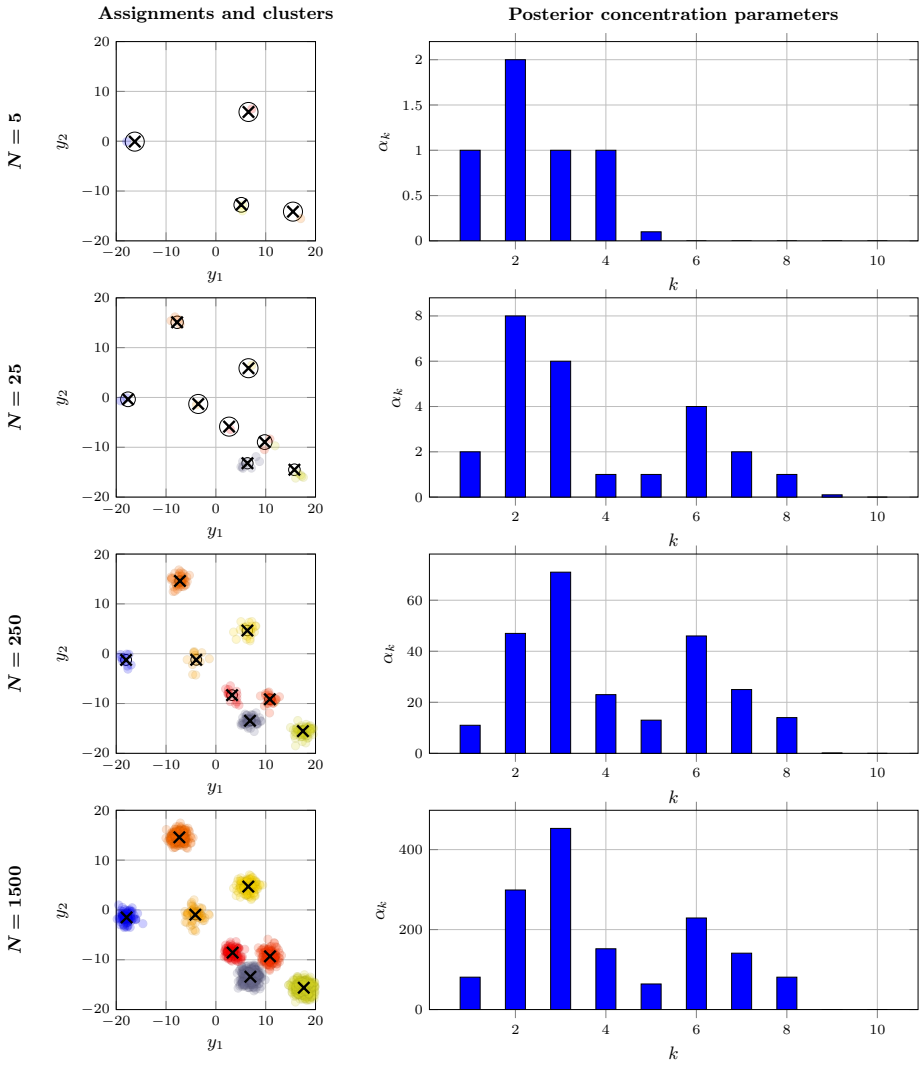


Fig. 3: Visualization of the results obtained by performing online inference in the infinite mixture model defined in Section 3 using the message passing implementation as described in Section 4. Each row represents the number of observations N. The left column shows the observations colored by their inferred cluster label and the inferred component means, denoted by square crosses. The right column denotes the concentration parameters of the approximate posterior distribution $q_N(\pi)$.

have effectively extended the task of model comparison to model expansion. The use of Dirichlet processes guarantees a well-grounded and principled approach

to the task of model expansion without any post-hoc treatment. A benefit of the online nature of the algorithm is that it is well-suited to the development of intelligent agents, which continuously perceive streams of information. Due to the online nature of the algorithm, its behavior also naturally depends on the ordering of the observations it perceives [36]. This is in contrast to sampling-based methods, however, these are significantly more computationally demanding.

It is important to note that the presented mixture node does not enforce any constraints on adjacent parts of the graph and can be used in both discrete and continuous spaces. A limitation of scale factors is that they can only be efficiently computed when the model submits to exact inference [27]. Extensions of the scale factors towards a variational setting would allow the use of the mixture node with a bigger variety of models. If this limitation is resolved, then the introduced approach can be combined with more complicated models, such as, for example, Bayesian neural networks, whose performance is measured by the variational free energy; see, e.g. [7, 17]. This provides a novel solution to multi-task machine learning problems where the number of tasks is not known beforehand [30]. Each Bayesian neural network can then be trained for a specific task, and additional components or networks can be added if appropriate.

Acknowledgments. The authors would like to thank the BIASlab team members for various insightful discussions related to this work. This publication is part of the project "ROBUST: Trustworthy AI-based Systems for Sustainable Growth" with project number KICH3.LTP.20.006, which is (partly) financed by the Dutch Research Council (NWO), GN Hearing, and the Dutch Ministry of Economic Affairs and Climate Policy (EZK) under the program LTP KIC 2020-2023.

Disclosure of Interests. The authors declare no conflict of interest.

References

1. Bagaev, D., van Erp, B., Podusenko, A., de Vries, B.: ReactiveMP.jl: A Julia package for reactive variational Bayesian inference. Software Impacts **12**, 100299 (May 2022). https://doi.org/10.1016/j.simpa.2022.100299, https://www.sciencedirect.com/science/article/pii/S2665963822000422
2. Bagaev, D., Podusenko, A., De Vries, B.: RxInfer: A Julia package for reactive real-time Bayesian inference. Journal of Open Source Software **8**(84), 5161 (Apr 2023). https://doi.org/10.21105/joss.05161, https://joss.theoj.org/papers/10.21105/joss.05161
3. Bagaev, D., de Vries, B.: Reactive Message Passing for Scalable Bayesian Inference. Scientific Programming **2023**, 6601690 (May 2023). https://doi.org/10.1155/2023/6601690, https://doi.org/10.1155/2023/6601690, publisher: Hindawi
4. Beckers, J., van Erp, B., Zhao, Z., Kondrashov, K., de Vries, B.: Principled Pruning of Bayesian Neural Networks Through Variational Free Energy Minimization. IEEE Open Journal of Signal Processing **5**, 195–203 (2024). https://doi.org/10.1109/OJSP.2023.3337718, https://ieeexplore.ieee.org/document/10334001, conference Name: IEEE Open Journal of Signal Processing

5. Bezanson, J., Edelman, A., Karpinski, S., Shah, V.B.: Julia: A Fresh Approach to Numerical Computing. SIAM Review **59**(1), 65–98 (Jan 2017). https://doi.org/10.1137/141000671, https://epubs.siam.org/doi/10.1137/141000671, publisher: Society for Industrial and Applied Mathematics
6. Blei, D.M., Jordan, M.I.: Variational inference for Dirichlet process mixtures. Bayesian Analysis **1**(1) (Mar 2006). https://doi.org/10.1214/06-BA104, https://projecteuclid.org/journals/bayesian-analysis/volume-1/issue-1/Variational-inference-for-Dirichlet-process-mixtures/10.1214/06-BA104.full
7. Blundell, C., Cornebise, J., Kavukcuoglu, K., Wierstra, D.: Weight Uncertainty in Neural Networks (May 2015). https://doi.org/10.48550/arXiv.1505.05424, http://arxiv.org/abs/1505.05424, arXiv:1505.05424 [cs, stat]
8. Dauwels, J., Korl, S., Loeliger, H.A.: Expectation maximization as message passing. In: Proceedings. International Symposium on Information Theory, 2005. ISIT 2005. pp. 583–586. IEEE, Adelaide, Australia (2005). https://doi.org/10.1109/ISIT.2005.1523402, http://ieeexplore.ieee.org/document/1523402/
9. van Erp, B., Nuijten, W.W.L., van de Laar, T., de Vries, B.: Automating Model Comparison in Factor Graphs. Entropy **25**(8), 1138 (Aug 2023). https://doi.org/10.3390/e25081138, https://www.mdpi.com/1099-4300/25/8/1138, number: 8 Publisher: Multidisciplinary Digital Publishing Institute
10. Ferguson, T.S.: A Bayesian Analysis of Some Nonparametric Problems. The Annals of Statistics **1**(2), 209–230 (Mar 1973). https://doi.org/10.1214/aos/1176342360, https://projecteuclid.org/journals/annals-of-statistics/volume-1/issue-2/A-Bayesian-Analysis-of-Some-Nonparametric-Problems/10.1214/aos/1176342360.full, publisher: Institute of Mathematical Statistics
11. Forney, G.: Codes on graphs: normal realizations. IEEE Transactions on Information Theory **47**(2), 520–548 (Feb 2001). https://doi.org/10.1109/18.910573
12. Fox, E.B., Sudderth, E.B., Jordan, M.I., Willsky, A.S.: An HDP-HMM for systems with state persistence. In: Proceedings of the 25th international conference on Machine learning. pp. 312–319. ICML '08, Association for Computing Machinery, New York, NY, USA (Jul 2008). https://doi.org/10.1145/1390156.1390196, https://dl.acm.org/doi/10.1145/1390156.1390196
13. Fox, E.B., Sudderth, E.B., Jordan, M.I., Willsky, A.S.: A Sticky Hdp-Hmm with Application to Speaker Diarization. The Annals of Applied Statistics **5**(2A), 1020–1056 (2011), https://www.jstor.org/stable/23024915, publisher: Institute of Mathematical Statistics
14. Friston, K., Parr, T., Zeidman, P.: Bayesian model reduction. arXiv:1805.07092 [stat] (Oct 2019), http://arxiv.org/abs/1805.07092, arXiv: 1805.07092
15. Friston, K., Penny, W.: Post hoc Bayesian model selection. NeuroImage **56**(4), 2089–2099 (Jun 2011). https://doi.org/10.1016/j.neuroimage.2011.03.062, http://www.sciencedirect.com/science/article/pii/S1053811911003417
16. Friston, K.J., Da Costa, L., Tschantz, A., Kiefer, A., Salvatori, T., Neacsu, V., Koudahl, M., Heins, C., Sajid, N., Markovic, D., Parr, T., Verbelen, T., Buckley, C.L.: Supervised structure learning (Nov 2023). https://doi.org/10.48550/arXiv.2311.10300, http://arxiv.org/abs/2311.10300, arXiv:2311.10300 [cs]
17. Haussmann, M., Hamprecht, F.A., Kandemir, M.: Sampling-Free Variational Inference of Bayesian Neural Networks by Variance Backpropagation (Jun 2019). https://doi.org/10.48550/arXiv.1805.07654, http://arxiv.org/abs/1805.07654
18. Hoeting, J.A., Madigan, D., Raftery, A.E., Volinsky, C.T.: Bayesian Model Averaging: A Tutorial. Statistical Science **14**(4), 382–401 (1999), https://www.jstor.org/stable/2676803

19. Kschischang, F., Frey, B., Loeliger, H.A.: Factor graphs and the sum-product algorithm. IEEE Transactions on Information Theory **47**(2), 498–519 (Feb 2001). https://doi.org/10.1109/18.910572
20. Lo, A.Y.: On a Class of Bayesian Nonparametric Estimates: I. Density Estimates. The Annals of Statistics **12**(1), 351–357 (1984), https://www.jstor.org/stable/2241054, publisher: Institute of Mathematical Statistics
21. Loeliger, H.A.: An introduction to factor graphs. IEEE Signal Processing Magazine **21**(1), 28–41 (Jan 2004). https://doi.org/10.1109/MSP.2004.1267047
22. Loeliger, H.A., Dauwels, J., Hu, J., Korl, S., Ping, L., Kschischang, F.R.: The Factor Graph Approach to Model-Based Signal Processing. Proceedings of the IEEE **95**(6), 1295–1322 (Jun 2007). https://doi.org/10.1109/JPROC.2007.896497, http://ieeexplore.ieee.org/document/4282128/
23. Lu, X., Zhang, C., Wang, Z.: Combined Belief Propagation-Mean Field Message Passing Algorithm for Dirichlet Process Mixtures. IEEE Signal Processing Letters **26**(7), 1041–1045 (Jul 2019). https://doi.org/10.1109/LSP.2019.2918680, https://ieeexplore.ieee.org/document/8721567, conference Name: IEEE Signal Processing Letters
24. Minka, T.P.: Expectation Propagation for Approximate Bayesian Inference. In: Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence. pp. 362–369. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2001), http://dl.acm.org/citation.cfm?id=2074022.2074067
25. Monteith, K., Carroll, J.L., Seppi, K., Martinez, T.: Turning Bayesian model averaging into Bayesian model combination. In: The 2011 International Joint Conference on Neural Networks. pp. 2657–2663. San Jose, CA, USA (Jul 2011). https://doi.org/10.1109/IJCNN.2011.6033566, iSSN: 2161-4407
26. Neal, R.M.: Markov Chain Sampling Methods for Dirichlet Process Mixture Models. Journal of Computational and Graphical Statistics **9**(2), 249–265 (Jun 2000). https://doi.org/10.1080/10618600.2000.10474879, https://www.tandfonline.com/doi/abs/10.1080/10618600.2000.10474879, publisher: Taylor & Francis _eprint: https://www.tandfonline.com/doi/pdf/10.1080/10618600.2000.10474879
27. Nguyen, H.M., van Erp, B., Şenöz, İ., de Vries, B.: Efficient Model Evidence Computation in Tree-structured Factor Graphs. In: 2022 IEEE Workshop on Signal Processing Systems (SiPS). pp. 1–6 (Nov 2022). https://doi.org/10.1109/SiPS55645.2022.9919250, iSSN: 2374-7390
28. Rasmussen, C.: The Infinite Gaussian Mixture Model. In: Advances in Neural Information Processing Systems. vol. 12. MIT Press (1999), https://papers.nips.cc/paper/1999/hash/97d98119037c5b8a9663cb21fb8ebf47-Abstract.html
29. Reller, C.: State-space methods in statistical signal processing: New ideas and applications. Ph.D. thesis, ETH Zurich (2013), http://hdl.handle.net/20.500.11850/65488
30. Ruder, S.: An Overview of Multi-Task Learning in Deep Neural Networks (Jun 2017), http://arxiv.org/abs/1706.05098, arXiv:1706.05098
31. Smith, R., Schwartenbeck, P., Parr, T., Friston, K.J.: An Active Inference Approach to Modeling Structure Learning: Concept Learning as an Example Case. Frontiers in Computational Neuroscience **14** (2020). https://doi.org/10.3389/fncom.2020.00041, https://www.frontiersin.org/articles/10.3389/fncom.2020.00041/full, publisher: Frontiers
32. Särkkä, S.: Bayesian Filtering and Smoothing. Institute of Mathematical Statistics Textbooks, Cambridge University Press, Cambridge (2013). https://doi.

org/10.1017/CBO9781139344203, https://www.cambridge.org/core/books/bayesian-filtering-and-smoothing/C372FB31C5D9A100F8476C1B23721A67
33. Teh, Y.W., Jordan, M.I.: Hierarchical Bayesian nonparametric models with applications. In: Bayesian Nonparametrics, pp. 158–207. Cambridge University Press, 1 edn. (Apr 2010). https://doi.org/10.1017/CBO9780511802478.006, https://www.cambridge.org/core/product/identifier/CBO9780511802478A043/type/book_part
34. Wang, L., Dunson, D.B.: Fast Bayesian Inference in Dirichlet Process Mixture Models. Journal of computational and graphical statistics : a joint publication of American Statistical Association, Institute of Mathematical Statistics, Interface Foundation of North America **20**(1), 10.1198/jcgs.2010.07081 (Jan 2011). https://doi.org/10.1198/jcgs.2010.07081, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3812957/
35. Winn, J.M.: Variational Message Passing and its Applications. Ph.D. thesis, University of Cambridge, Cambridge, United Kingdom (2004)
36. Zhang, X., Nott, D.J., Yau, C., Jasra, A.: A Sequential Algorithm for Fast Fitting of Dirichlet Process Mixture Models. Journal of Computational and Graphical Statistics **23**(4), 1143–1162 (2014), https://www.jstor.org/stable/43304802, publisher: [American Statistical Association, Taylor & Francis, Ltd., Institute of Mathematical Statistics, Interface Foundation of America]
37. Şenöz, İ., van de Laar, T., Bagaev, D., de Vries, B.: Variational Message Passing and Local Constraint Manipulation in Factor Graphs. Entropy **23**(7), 807 (Jul 2021). https://doi.org/10.3390/e23070807, https://www.mdpi.com/1099-4300/23/7/807, number: 7 Publisher: Multidisciplinary Digital Publishing Institute

Reactive Environments for Active Inference Agents with RxEnvironments.jl

Wouter W.L. Nuijten[1][0009−0007−0689−9300] and Bert de Vries[1,2][0000−0003−0839−174X] {w.w.l.nuijten, bert.de.vries}@tue.nl

[1] Eindhoven University of Technology, 5612 AP Eindhoven, the Netherlands
[2] GN Hearing, 5612 AB Eindhoven, The Netherlands

Abstract. Active Inference is a framework that emphasizes the interaction between agents and their environment. While the framework has seen significant advancements in the development of agents, the environmental models are often borrowed from reinforcement learning problems, which may not fully capture the complexity of multi-agent interactions or allow complex, conditional communication. This paper introduces Reactive Environments, a comprehensive paradigm that facilitates complex multi-agent communication. In this paradigm, both agents and environments are defined as entities encapsulated by boundaries with interfaces. This setup facilitates a robust framework for communication in nonequilibrium-Steady-State systems, allowing for complex interactions and information exchange. We present a Julia package `RxEnvironments.jl`, which is a specific implementation of Reactive Environments, where we utilize a Reactive Programming style for efficient implementation. The flexibility of this paradigm is demonstrated through its application to several complex, multi-agent environments. These case studies highlight the potential of Reactive Environments in modeling sophisticated systems of interacting agents.

Keywords: Active Inference · Agent-Environment Interaction · Reactive Environments · Reactive Programming

1 Introduction

The Free Energy Principle (FEP) [7] distinguishes itself from other theories of self-organization by taking an interaction-centric perspective. Active inference (AIF) is an implication of the Free Energy Principle that extends the FEP to control and decision-making in self-organizing natural systems.

In this framework, agents possess an internal generative model for predicting observations from an unknown external process. The model updates its internal (perceptive) and active (control) states to minimize prediction errors. This unifying principle has profound implications for understanding how agents perceive and act within complex environments.

AIF posits that agents actively seek to minimize their free energy (a measure related to surprise or prediction error) by updating their beliefs about the environment and selecting actions that align with these beliefs [9]. This formulation bridges the gap between theoretical principles and practical implementations of FEP in agent-environment interactions.

To simulate a synthetic AIF agent, researchers need the ability to control interactions between agents and their environment in practical scenarios. For example, a significant theory from AIF is that the human brain learns from the proprioceptive feedback it receives from muscles [1]. Since proprioceptive and exteroceptive sensory channels do not necessarily run at the same time-frequency rates, researchers need fine-grained control over the communication protocol between the agent and the environment. In other settings, one could be interested in having multiple agents share the same world, allowing communication between agents [10]. Current solutions from the reinforcement learning or control theory community, such as Gymnasium [23] do not give end-users these controls over details of the environment, instead focusing on implementing a single agent-environment interaction through a transition function. The imperative programming style used in these frameworks limits the communication between agents and environments with a predefined time step, observation frequency, and action frequency.

This paper introduces Reactive Environments, which adopt a reactive programming approach to environment design. In contrast to their imperative counterparts, Reactive Environments are not limited by strict communication constraints and natively allow multi-sensor, multimodal interaction between agent and environment. We will discuss how a reactive programming strategy addresses the flaws of current frameworks and introduce *RxEnvironments.jl*, a specific implementation of Reactive Environments in the Julia language [4]. We will show how implementing complex real-world environments with fine-grained control over an agent's observations is streamlined in RxEnvironments.jl. The main features of RxEnvironments.jl are:

- Detailed control over observations. Different sensory channels can execute at different frequencies or can be triggered only when specific actions are taken, allowing for complex interactions.
- Native support for multi-agent environments: multiple instances of the same agent type can be spawned in the same environment without additional code.
- Reactivity: By employing a reactive programming style, we ensure that environments will emit observations when prompted, and will idle when no computation is necessary.
- Support for multi-entity complex environments where the agent-environment framework does not suffice.

With RxEnvironments, we hope to contribute to standardizing the creation and simulation of Active Inference agents, allowing researchers to share their environments and potentially creating standardized benchmarks in the future.

The main contributions of this paper are as follows:

- We define the Reactive Environment concept in Section 3.2.
- In Section 3.3, we introduce RxEnvironments as a package to create environments for Active Inference agents.
- In Section 4, we demonstrate how to create complex environments with unique needs.

2 Related Work

In reinforcement learning, the creation and sharing of control environments has mainly been standardized with the introduction of Gymnasium [23]. Users can use the step function in Gym to define a transition function, and Gym will handle the environmental simulation. A similar alternative in Python, based on the MuJoCo physics engine, is Deepmind Control Suite [21]. The equivalent alternative in the Julia programming language would be ReinforcementLearning.jl [22]. These packages export high-level interfaces to the environments they describe, alleviating the user's burden of timekeeping. Although these packages are designed explicitly for reinforcement learning, which involves computing a reward metric at every state, they can also be used for Active Inference as they describe general control environments [24,17]. Although the realization of environments is also part of popular packages such as PyMDP [11] and the SPM-DEM toolbox [6], these packages have their primary focus on agent creation. As a result, we do not present RxEnvironments as a substitute but rather as a comprehensive framework that agent-centric packages can use.

In general, we observe that there is no standardized way of defining environments for Active Inference agents. While some implementations use Gymnasium [24,17,20], others use specialized toolboxes for implementing Active Inference agents for their environment simulation [5,10]. With RxEnvironments, we aim to unite all use cases for environments in Active Inference in a robust and comprehensive package.

Reactive Programming (RP) has wide applications in various domains and is similar to the Actor Model [13]. RP does not assume anything about the data generation process, allowing computation both on static datasets and real-time asynchronous sensor observations. In Reactive Programming, it is necessary to define how the system should react to changes in data or events rather than explicitly programming sequences of steps. This approach is similar to an in situ control system that can gather data through its sensors asynchronously and respond accordingly to incoming stimuli.

3 Methods

3.1 A Model for Interaction in Active Inference

Self-organizing agents maintain their existence by creating a boundary that separates their internal states from the external states of their environment [12,25,18,15]. An agent can only affect its environment through its actuators,

while its internal states are influenced only by stimuli received through its sensors. Therefore, an agent has a set of actuator and sensor interfaces that it uses to communicate with its environment. We will refer to this collection of all actuators and sensors of an agent as its boundary. A schematic of this interaction model can be seen in Figure 1.

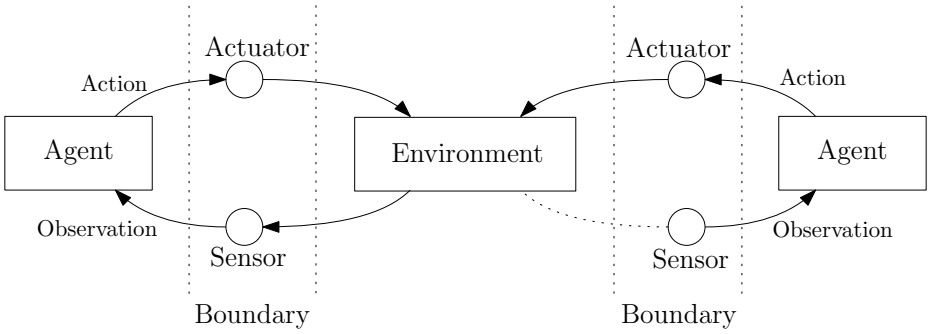


Fig. 1. General communication protocol in an Active Inference environment containing two agents. The terms "Actuator" and "Sensor" are used from the agents' points of view. We see that both agents have a boundary with actuators and sensors with which they interact with the environment.

Here, the duality between the agent and environment is notable; the actions emitted by the agent are perceived as observations for the environment, and vice versa. This dual separation of internal and external states has prompted an overarching term for agents and environments, which we call an **Entity**. Note that Entities are separate from Fristonian "things" [8] in the sense that an Entity can be any "thing" or an environment, making it a superclass to things. We use the following definition of an Entity:

Definition 1. *Entity. An Entity is a structure with a set of actuators and sensors called a boundary that allows it to communicate with other Entities.*

An Entity can, but is not obliged to, have an internal state that the sensor interfaces in its boundary can influence. Entities with connected pairs of actuators and sensors are mutually "subscribed." We use this term because any action emitted by an agent prompts a change in the internal state of an environment. Generally, an emitted action by an Entity prompts activity (either a response or an update of an internal state) from the subscribed entities. In Figure 1, we see three entities, two agents that are both subscribed to the same environment Entity, with the boundaries of both agents expanded.

The notion of a Markov Blanket is prevalent in Active Inference literature [18,14], and it denotes the statistical partition between a system's internal and external states. It is worth noting that the concept of a boundary, as formulated in this section, coincides with the notion of a Markov Blanket used in Active Inference. Therefore, modeling communication as an interaction between different

Entities through their boundaries is an adequate implementation of the communication of a probabilistic model with a Markov Blanket and its environment.

3.2 Reactive Environments

Communication between Entities flows through their respective boundaries. Any Entity can send data through its actuator interface to its subscribers at any point, prompting activity in subscribed entities. Building upon the discussion in Section 2 regarding the Reactive Programming paradigm, we extend the concept to environments in agent-based systems. Entities should process sensory data seamlessly and respond to subscribers. Just as programmers define how systems respond to impulses, we aim to define how entities react to impulses exerted by the entities to which they are subscribed. To this extent, we define a Reactive Environment:

Definition 2. *Reactive Environments. A Reactive Environment is a pair* $\mathcal{E} = (\mathcal{A}, \mathcal{S})$ *where* $\mathcal{A}$ *is a set of Entities and* $\mathcal{S}$ *is a mutable set of subscriptions, where every* $s \in \mathcal{S}$ *is a pair* $(\mathcal{A}_1, \mathcal{A}_2)$*,* $\mathcal{A}_2, \mathcal{A}_2 \in \mathcal{A}$*. Each Entity responds reactively to sensory impulses received from any of its subscribers through its sensors and is able to emit data to any subscriber through its actuators.*

For Entities in a Reactive Environment, we provide a set of desiderata that enable the design of complex communication networks within the Reactive Environment framework.

- Entities should be able to update their internal state in response to received impulses. The update should be based either on the impulse emitter or the type of observation. For example, an agent should update its internal state differently when receiving an audio signal versus a video signal.
- An Entity should be able to determine whether or not to transmit any received impulse to its subscribers. For example, suppose an agent sends a proprioceptive signal, and the Entity representing the environment receives it. In that case, the environment can match it with an observation, but it does not have to transmit it to its other subscribers.
- At any given time, an Entity should be able to send a signal to its subscribers, such as a video camera emitting a 60Hz signal.
- An Entity should be able to send different signals to different subscribers when emitting. For example, an environment Entity can send different observations to different agent entities based on their relative position in the environment.

In Figure 2, we see a flow chart of the logic we want every Entity to go through whenever they get an observation. This logic can also be triggered regularly to mimic a sensor continuously providing data at a fixed rate. An algorithm implementing the logic in this flowchart allows for all the behaviors listed above.

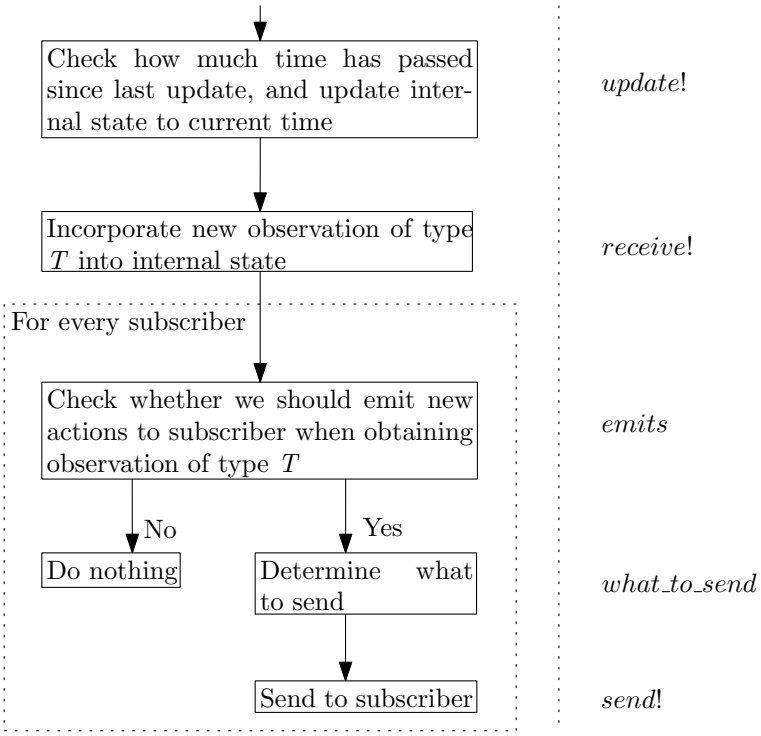


Fig. 2. Internal Entity logic is applied when an observation is received. On the left, we outline the steps an Entity should follow when processing an observation. On the right, we specify the `RxEnvironments` functions that users can create to customize this behavior.

3.3 RxEnvironments.jl: a Particular Implementation of Reactive Environments

In this section, we introduce `RxEnvironments.jl`[3], a package in the open-source Julia [4] language that implements the communication protocol and the desiderata described in Section 3.2. In RxEnvironments, we take an Entity-centric standpoint and implement all communication logic on the Entity level. This means that entities representing agents and environments have no constraints on their communication, allowing agent-agent subscriptions or multi-agent environments natively. The reactive programming features of RxEnvironments are based on the *Rocket.jl* Reactive Programming library [2]. In Figure 2 we see the flow chart that describes the logic entities go through when processing observations, on the right we see the corresponding functions in `RxEnvironments`. In this section, we will go through these functions in more detail.

[3] https://github.com/biaslab/RxEnvironments.jl

Revising the transition function In popular environment design frameworks, such as Gymnasium [23], the transition function for an environment takes the action emitted by an agent and the previous state. It produces a new state and an observation for the agent. In our Entity-centric approach, this modeling assumption is restrictive for various reasons:

- In a scenario where entities can have multiple subscribers and be subscribed to multiple entities, it is not desirable to trigger the transition function for every received stimulus. To illustrate, consider a multi-agent environment where all actions emitted by each agent should be gathered before updating and advancing the environment time. This ensures that the transition function is not invoked too frequently.
- The transition function cannot have a constant time interval. It should be a function of the elapsed time since the last update to account for stimuli being received at any time.

For these reasons, we split the transition update function into two different distinct parts:

1. The function `receive!` updates the internal state of the receiving Entity by incorporating the stimulus received, given the emitting and receiving entities and the stimulus as inputs.
2. The function `update!` takes an Entity and the time elapsed since the last update of this Entity and updates the internal state of the Entity to reflect that additional time has elapsed.

By splitting these two functions, we can natively support multi-agent environments, and we can simulate the behavior of the internal state of an Entity in the absence of stimuli. A concrete example would be that multiple agents could still observe a ball bouncing in an environment without explicitly abstaining from action at every time step since the dynamics of the bouncing ball is described in the `update!` function, which will continue to be called even in the absence of stimuli.

Control over emission logic In Section 3.2, we have emphasized the importance of controlling the emission logic when an Entity receives a stimulus. Therefore, we expose two functions that govern emission logic for entities in RxEnvironments:

1. The `emits` function operates by taking in the receiving Entity, the stimulus data received by the Entity, and any subscribers of the Entity. The function then returns a value determining whether the Entity should emit the stimulus to a particular subscriber. This function is called for all subscribers.
2. The function `what_to_send` takes the receiving Entity, the stimulus data that the Entity receives, and any subscriber of the Entity. This function is called when `emits` returns true and determines which stimulus will be sent to that subscriber.

In this way, designers of environments have full control over when and how their entities should emit. In the next section, we will demonstrate the versatility of this framework by designing several complex environments.

Internal triggers for emission Of course, not all emissions in Entity interactions are triggered by external impulses. Sensors, for example, usually provide data at a steady frequency. Therefore, we expose an interface to emit observations to all subscribers at regular intervals. By creating entities for different sensors and attaching them to an overarching Entity, we can create complex systems that combine data from different sensors at different observation frequencies.

Replicating classical reinforcement learning environments In classical reinforcement learning environments, the environment is often seen as a passive recipient of actions from the agent, responding to actions with observations in the next time step. A different way to view environments is to consider them as reactive environments. In this case, the Entity representing the environment waits until all subscribed entities have emitted before calling the state transition function at a predetermined time interval. This approach transforms classical reinforcement learning environments into specific instances of reactive environments, making environmental simulation more flexible and generalizable.

4 Case Studies

In this section, we will implement several increasingly complex environments. In every environment, we employ a specific strategy from Reactive Environments that cannot be replicated in popular environment creation packages. [4]

4.1 Mountain Car

In this section, we implement a classic environment in reinforcement learning, the Mountain Car environment [24], with a slight difference: whenever an agent emits an action (e.g., setting the throttle of the engine), we match this with an observation from the environment that contains the engine force applied. For example, if the agent decides to apply 300% of its engine force, the environment will reply that only 100% is applied. Presenting observations in this way aligns with [1]. To realize this, we design our environment to trigger different implementations of `what_to_send` based on the input stimulus: Whenever a throttle action is received, we return the throttle action that is applied to the environment, and we let the environment emit on a regular frequency of 2 Hz to replicate a sensor that measures the position and velocity of the mountain car. In Figure 3, we see a schematic overview of the interactions between the agent and the environment. We see that we have two distinct types of observations from the

[4] The implementation of the environments discussed in this section can be found at `https://github.com/wouterwln/RxEnvironments-Examples`

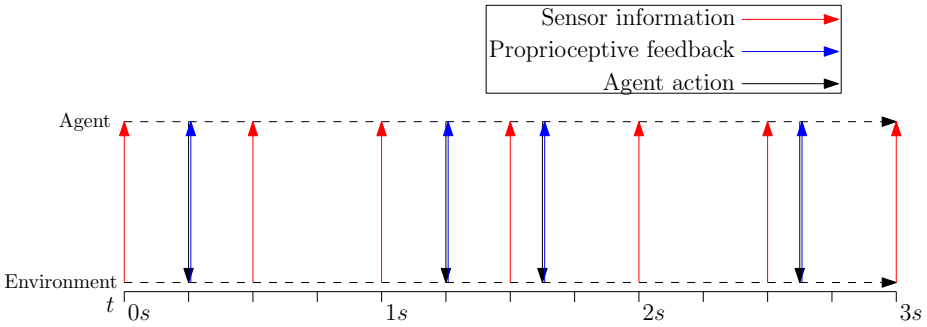


Fig. 3. Overview of interactions in the Mountain Car environment over time. The environment emits sensor information at a regular interval (2 Hz in this example), and whenever the agent emits an action, the environment instantaneously responds with proprioceptive feedback to the agent.

agent, sensory and proprioceptive feedback, and the logic for obtaining both is different.

4.2 Football Environment

Next, we describe the implementation of a simulation environment of a football match. Football is the most popular sport in the world and involves two teams of 11 players who handle a ball with their feet to score goals. A football game is a complex multi-agent game where the entity representing the environment has to handle inputs from all 22 agents to let the game run smoothly. Additionally, since we are in a noncooperative game between two teams, players can emit signals (shout) to their teammates, so we also have an agent-to-agent communication channel. Since all players have their position and orientation on the pitch, their field of vision and the observations they receive are also different for each agent.

We model this environment with a single Entity representing the state of the world and 22 Entities representing the individual players. The world contains the ball and the references to all 22 player bodies, so collisions and on-ball actions can be resolved. All player Entities are subscribed to the world Entity but are not subscribed to each other. We do not explicitly model agent-to-agent interactions because this would unnecessarily complicate the subscription graph. Instead, a player can choose to emit a signal to all other players, which the world Entity will forward to all other players. In a sense, the world Entity represents the "global" state of the system that keeps track of all physical interactions. At the same time, all player Entities contain their local states and receive observations from the global state. In Figure 4, we have visualized an example of this environment. This example shows that we can, with the code used to define a single player, create a 22-player environment. In this YouTube video, we show that we can send commands to all individual players asynchronously. Here, we send the command to run in a random direction to a random player every 0.1 seconds.

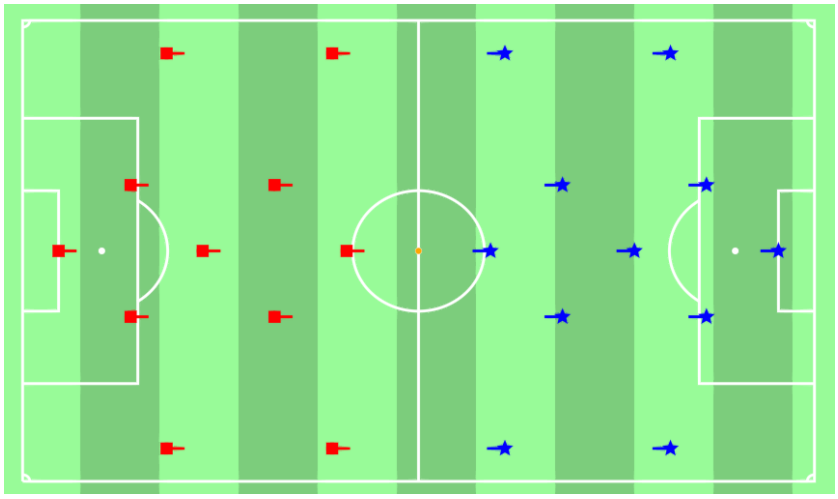


Fig. 4. Plot of the setup of our football environment, showing the pitch and the 22 players. The ball is positioned on the center spot. An animation of this environment where we send random run commands to players can be found here.

Due to the intricate dynamics of the football game and its non-trivial set of rules, we have decided only to model running and on-ball actions. We aim to demonstrate the multi-agent nature of Reactive Environments rather than create a comprehensive football environment.

4.3 Hearing Aid Environment

Hearing aids often feature advanced acoustic noise reduction algorithms. In recent years, we have seen the rise of active inference-based agents that parameterize hearing aid noise reduction algorithms [19]. Since a hearing aid has very limited computing power and battery capacity, sometimes part of the agent's needed computations must be performed on a separate wearable device, e.g., the patient's phone. This configuration leads to a unique multi-entity system where the hearing aid is continually communicating with three different entities: (1) the outside world, which emits acoustic signals; (2) the user (hearing aid patient), who receives the hearing aid output signal (and can potentially emit feedback to the hearing aid about the perceived performance of the hearing aid); and (3) with the intelligent agent at the user's phone. Figure 5 shows a schematic overview.

Thus, we obtain a complex entity interaction, where the hearing aid can obtain stimuli from all 3 subscribed entities and should process the data accordingly: an acoustic signal from the outside world should be processed and emitted to both the user and the agent; a new proposal for parameter settings, i.e., the agent's actions, should be incorporated into the signal processing algo-

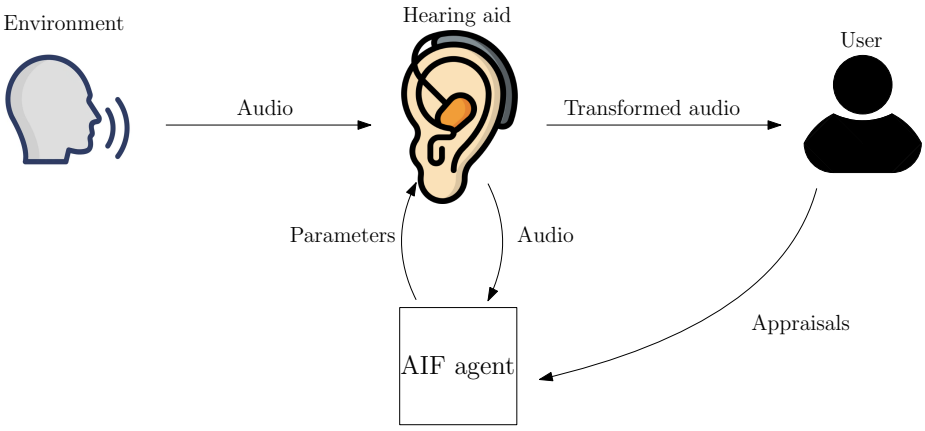


Fig. 5. Schematic of the subscriptions in the hearing aid environment.

rithm; and user appraisals should also be forwarded to the agent that will use these appraisals to update its future parameter proposals.

In short, we have a complex multi-entity interaction where every entity should handle different stimuli in different ways. We have to control which signal to emit and when to emit the signal. Note that the hearing aid should not send a signal to the user when receiving a new set of parameters but only when the sound from the outside environment is registered. In a Reactive Environment, all interactions are well-defined, and we can observe the signal the user hears while also designing an agent that would take the place of the agent in our setting.

5 Discussion

While Reactive Environments generalize environments for learning agents, the design of agents that can interact with a Reactive Environment should still be investigated. This is because, in our framework, there are fewer constraints on the communication between an agent and its surroundings. Traditionally, our agent receives 1 observation per predetermined timestep and can process this observation accordingly. In a Reactive Environment, agents can receive (or not receive) data at any point in time, necessitating an internal clock for the agents themselves. Although relieving this constraint on communication allows for interesting experiments (we can disable a sensor for a particular agent to simulate the sensor breaking down and investigate how our agent handles this loss of data), the authors are not aware of any implementation of agents that are built for this level of flexibility. An interesting avenue is Reactive Message Passing on a Factor Graph [16,3], which employs the same reactive programming strategy to Bayesian inference that we have taken in this paper to environment design.

6 Conclusions

In this paper, we presented the concept of a Reactive Environment and a particular implementation, `RxEnvironments.jl`. We showed that environments defined in the classical reinforcement learning literature can be written as particular cases of Reactive Environments, and we showed that we can model more complex interactions within this paradigm. In particular, we showed that with Reactive Environments we are able to model the complex communications between agents and environments necessary to realize Active Inference simulations. In our case studies, we showed that our framework can be used to define a multitude of different environments, demonstrating the expressive power of the framework. Furthermore, we have presented `RxEnvironments.jl`, a particular implementation of Reactive Environments. Extensions of this work might investigate the classes of agents that handle the communication protocol employed by Reactive Agents to simulate how agents would operate in the field.

Acknowledgements

This publication is part of the project "ROBUST: Trustworthy AI-based Systems for Sustainable Growth" with project number KICH3.LTP.20.006, which is (partly) financed by the Dutch Research Council (NWO), GN Hearing, and the Dutch Ministry of Economic Affairs and Climate Policy (EZK) under the program LTP KIC 2020-2023.

The authors thank Thijs van de Laar, Magnus Koudahl, and Tim Nisslbeck for their insightful discussions during the project's execution.

References

1. Adams, R.A., Shipp, S., Friston, K.J.: Predictions not commands: active inference in the motor system. Brain Structure & Function **218**(3), 611–643 (2013). https://doi.org/10.1007/s00429-012-0475-5, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3637647/
2. Bagaev, D.: Rocket.jl: Reactive extensions library for Julia (2020), https://github.com/ReactiveBayes/Rocket.jl
3. Bagaev, D., de Vries, B.: Reactive Message Passing for Scalable Bayesian Inference (Dec 2021). https://doi.org/10.48550/arXiv.2112.13251, http://arxiv.org/abs/2112.13251, arXiv:2112.13251 [cs]
4. Bezanson, J., Edelman, A., Karpinski, S., Shah, V.B.: Julia: A Fresh Approach to Numerical Computing (Jul 2015). https://doi.org/10.48550/arXiv.1411.1607, http://arxiv.org/abs/1411.1607, arXiv:1411.1607 [cs]
5. Esaki, K., Matsumura, T., Minusa, S., Shao, Y., Yoshimura, C., Mizuno, H.: Dynamical Perception-Action Loop Formation with Developmental Embodiment for Hierarchical Active Inference. In: Buckley, C.L., Cialfi, D., Lanillos, P., Ramstead, M., Sajid, N., Shimazaki, H., Verbelen, T., Wisse, M. (eds.) Active Inference. pp. 14–28. Springer Nature Switzerland, Cham (2024). https://doi.org/10.1007/978-3-031-47958-8_2

6. Friston, K.J., Trujillo-Barreto, N., Daunizeau, J.: DEM: a variational treatment of dynamic systems. NeuroImage **41**(3), 849–885 (Jul 2008). https://doi.org/10.1016/j.neuroimage.2008.02.054
7. Friston, K.: The free-energy principle: a unified brain theory? Nature Reviews Neuroscience **11**(2), 127–138 (Feb 2010). https://doi.org/10.1038/nrn2787, https://www.nature.com/articles/nrn2787, number: 2 Publisher: Nature Publishing Group
8. Friston, K.: A free energy principle for a particular physics (Jun 2019). https://doi.org/10.48550/arXiv.1906.10184, http://arxiv.org/abs/1906.10184, arXiv:1906.10184 [q-bio]
9. Friston, K.J., Daunizeau, J., Kilner, J., Kiebel, S.J.: Action and behavior: a free-energy formulation. Biological Cybernetics **102**(3), 227–260 (Mar 2010). https://doi.org/10.1007/s00422-010-0364-z, https://doi.org/10.1007/s00422-010-0364-z
10. Friston, K.J., Parr, T., Heins, C., Constant, A., Friedman, D., Isomura, T., Fields, C., Verbelen, T., Ramstead, M., Clippinger, J., Frith, C.D.: Federated inference and belief sharing. Neuroscience & Biobehavioral Reviews **156**, 105500 (Jan 2024). https://doi.org/10.1016/j.neubiorev.2023.105500, https://www.sciencedirect.com/science/article/pii/S0149763423004694
11. Heins, C., Millidge, B., Demekas, D., Klein, B., Friston, K., Couzin, I.D., Tschantz, A.: pymdp: A Python library for active inference in discrete state spaces. Journal of Open Source Software **7**(73), 4098 (May 2022). https://doi.org/10.21105/joss.04098, https://joss.theoj.org/papers/10.21105/joss.04098
12. Hesp, C., Ramstead, M., Constant, A., Badcock, P., Kirchhoff, M., Friston, K.: A Multi-scale View of the Emergent Complexity of Life: A Free-Energy Proposal. In: Georgiev, G.Y., Smart, J.M., Flores Martinez, C.L., Price, M.E. (eds.) Evolution, Development and Complexity. pp. 195–227. Springer Proceedings in Complexity, Springer International Publishing, Cham (2019). https://doi.org/10.1007/978-3-030-00075-2_7
13. Hewitt, C., Bishop, P., Steiger, R.: Session 8 formalisms for artificial intelligence a universal modular actor formalism for artificial intelligence. In: Advance papers of the conference. vol. 3, p. 235. Stanford Research Institute Menlo Park, CA (1973)
14. Kaufmann, R., Gupta, P., Taylor, J.: An active inference model of collective intelligence. Entropy **23**(7), 830 (Jun 2021). https://doi.org/10.3390/e23070830, http://arxiv.org/abs/2104.01066, arXiv:2104.01066 [cs, eess]
15. Kirchhoff, M., Parr, T., Palacios, E., Friston, K., Kiverstein, J.: The Markov blankets of life: autonomy, active inference and the free energy principle. Journal of The Royal Society Interface **15**(138), 20170792 (Jan 2018). https://doi.org/10.1098/rsif.2017.0792, https://royalsocietypublishing.org/doi/10.1098/rsif.2017.0792, publisher: Royal Society
16. Loeliger, H.A., Dauwels, J., Hu, J., Korl, S., Ping, L., Kschischang, F.R.: The Factor Graph Approach to Model-Based Signal Processing. Proceedings of the IEEE **95**(6), 1295–1322 (Jun 2007). https://doi.org/10.1109/JPROC.2007.896497
17. Van de Maele, T., Dhoedt, B., Verbelen, T., Pezzulo, G.: Integrating cognitive map learning and active inference for planning in ambiguous environments (Aug 2023). https://doi.org/10.48550/arXiv.2308.08307, http://arxiv.org/abs/2308.08307, arXiv:2308.08307 [cs]

18. Palacios, E.R., Razi, A., Parr, T., Kirchhoff, M., Friston, K.: On Markov blankets and hierarchical self-organisation. Journal of Theoretical Biology **486**, 110089 (Feb 2020). https://doi.org/10.1016/j.jtbi.2019.110089, https://www.sciencedirect.com/science/article/pii/S0022519319304588
19. Podusenko, A., van Erp, B., Koudahl, M., de Vries, B.: AIDA: An Active Inference-Based Design Agent for Audio Processing Algorithms. Frontiers in Signal Processing **2** (Mar 2022). https://doi.org/10.3389/frsip.2022.842477, https://www.frontiersin.org/articles/10.3389/frsip.2022.842477, publisher: Frontiers
20. Safa, A., Verbelen, T., Keuninckx, L., Ocket, I., Bourdoux, A., Catthoor, F., Gielen, G., Cauwenberghs, G.: Active Inference in Hebbian Learning Networks (Jun 2023). https://doi.org/10.48550/arXiv.2306.05053, http://arxiv.org/abs/2306.05053, arXiv:2306.05053 [cs]
21. Tassa, Y., Doron, Y., Muldal, A., Erez, T., Li, Y., Casas, D.d.L., Budden, D., Abdolmaleki, A., Merel, J., Lefrancq, A., Lillicrap, T., Riedmiller, M.: DeepMind Control Suite (Jan 2018). https://doi.org/10.48550/arXiv.1801.00690, http://arxiv.org/abs/1801.00690, arXiv:1801.00690 [cs]
22. Tian, J.: ReinforcementLearning. jl: A reinforcement learning package for the julia programming language (2020), https://github.com/JuliaReinforcementLearning/ReinforcementLearning.jl
23. Towers, M., Terry, J.K., Kwiatkowski, A., Balis, J.U., Cola, G., Deleu, T., Goulão, M., Kallinteris, A., KG, A., Krimmel, M., Perez-Vicente, R., Pierré, A., Schulhoff, S., Tai, J.J., Tan, A.J.S., Younis, O.G.: Gymnasium (Feb 2024). https://doi.org/10.5281/zenodo.10655021, https://zenodo.org/records/10655021
24. Ueltzhöffer, K.: Deep Active Inference. Biological Cybernetics **112**(6), 547–573 (Dec 2018). https://doi.org/10.1007/s00422-018-0785-7, http://arxiv.org/abs/1709.02341, arXiv:1709.02341 [q-bio]
25. Varela, F.G., Maturana, H.R., Uribe, R.: Autopoiesis: The organization of living systems, its characterization and a model. Biosystems **5**(4), 187–196 (May 1974). https://doi.org/10.1016/0303-2647(74)90031-8, https://www.sciencedirect.com/science/article/pii/0303264774900318

A Appendix: Creating a simple environment

In this code example, we will demonstrate the creation of a simple environment in `RxEnvironments`, demonstrating that the additional boilerplate code needed to write an imperative environment as a Reactive Environment is minimal. We will implement the Bayesian Thermostat example, which is also showcased in the RxEnvironments documentation.

A.1 Defining the environment

The Bayesian Thermostat environment is a very simple environment that monitors the temperature in a room. The temperature can fluctuate between a minimal and a maximal temperature, and an agent can influence this temperature by adding or subtracting heat from the room. Furthermore, the environment cools down over time.

A.2 Environment boilerplate

In this section we will write all boilerplate code necessary to run the environment. We start by defining the structures needed to store the temperature and environment properties and expose helper functions that change this temperature, namely the `add_temperature!` function.

```
using Distributions

# Empty agent, could contain states as well
struct ThermostatAgent end

mutable struct BayesianThermostat{T}
    temperature::T
    min_temp::T
    max_temp::T
end

# Helper functions
temperature(env::BayesianThermostat) = env.temperature
min_temp(env::BayesianThermostat) = env.min_temp
max_temp(env::BayesianThermostat) = env.max_temp
noise(env::BayesianThermostat) = Normal(0.0, 0.1)
set_temperature!(env::BayesianThermostat, temp::Real) = env.
    temperature = temp
function add_temperature!(env::BayesianThermostat, diff::
    Real)
    env.temperature += diff
    if temperature(env) < min_temp(env)
        set_temperature!(env, min_temp(env))
    elseif temperature(env) > max_temp(env)
        set_temperature!(env, max_temp(env))
    end
end
```

A.3 RxEnvironments specific code

In this section, we will implement the `RxEnvironments`-specific code. We have a very simple interaction scheme: When the agent emits an action, we want this to be incorporated into the environment state, but we only want the environment to emit observations on a fixed frequency, and not present an observation whenever the agent chooses to change the environment. Therefore, we implement the `receive!`, `update!`, `emits` and `what_to_send` functions for the environment:

```
# When the environment receives an action from the agent, we
      shouldn't emit back to the agent
RxEnvironments.emits(::BayesianThermostat, ::ThermostatAgent
    , ::Real) = false

# In any other case, we should emit (This line is obsolete
    since this is the default behavior, but we include it
    for clarity)
RxEnvironments.emits(::BayesianThermostat, ::ThermostatAgent
    , any) = true

# When the environment receives an action from the agent, we
     add the value of the action to the environment
    temperature.
RxEnvironments.receive!(recipient::BayesianThermostat,
    emitter::ThermostatAgent, action::Real) =
    add_temperature!(recipient, action)

# The environment sends a noisy temperature observation to
    the agent.
RxEnvironments.what_to_send(recipient::ThermostatAgent,
    emitter::BayesianThermostat) = temperature(emitter) +
    rand(noise(emitter))

# The environment cools down over time.
RxEnvironments.update!(env::BayesianThermostat, elapsed_time
    )= add_temperature!(env, -0.1 * elapsed_time)
```

A.4 Invoking the environment

We now have all the code necessary to kickstart our environment:

```
environment = RxEnvironment(BayesianThermostat(0.0, -10.0,
    10.0); emit_every_ms = 1000)
agent = add!(environment, ThermostatAgent())
```

Now, your environment will be running, and `agent` will receive a noisy observation from the environment every second.

Modeling Sustainable Resource Management using Active Inference

Mahault Albarracin[1], Ines Hipolito[2], Maria Raffa[3], Paul Kinghorn[4]

[1]Université du Québec à Montréal, Montréal, Canada
[2]Macquarie University, Sydney, Australia
[3]IULM University, Milan, Italy
[4]Department of Informatics and Engineering, University of Sussex, Brighton, UK

Abstract. Active inference helps us simulate adaptive behavior and decision-making in biological and artificial agents. Building on our previous work exploring the relationship between active inference, well-being, resilience, and sustainability, we present a computational model of an agent learning sustainable resource management strategies in both static and dynamic environments. The agent's behavior emerges from optimizing its own well-being, represented by prior preferences, subject to beliefs about environmental dynamics. In a static environment, the agent learns to consistently consume resources to satisfy its needs. In a dynamic environment where resources deplete and replenish based on the agent's actions, the agent adapts its behavior to balance immediate needs with long-term resource availability. This demonstrates how active inference can give rise to sustainable and resilient behaviors in the face of changing environmental conditions. We discuss the implications of our model, its limitations, and suggest future directions for integrating more complex agent-environment interactions. Our work highlights active inference's potential for understanding and shaping sustainable behaviors.

Keywords: Active Inference · Sustainability · Generative model.

1 Introduction

For the past decade, we have shown that the Free Energy Principle can serve as a foundational concept in predicting and modeling the present and future behaviors of a system. Under this principle, the behavior of a system aims to maintain equilibrium and sustaining life through the minimization of free energy [Parr et al., 2022, Parr and Friston, 2019, Stubbs and Friston, 2024]. Systems, particularly biological ones, act to minimize the difference between their representation of the world, encoded in internal states and the external environment. By reducing this discrepancy, as quantified by free energy, the system achieves a state of balance and effectively adapts to its surroundings [Friston et al., 2017, 2023a, Parr et al., 2023]. We can thus understand it as a measure of uncertainty or

surprise, used such that agents are driven to more predictable and stable states. Using free energy, systems can slowly make adjustments and adaptations, and thus maintain homeostasis in a changing environment [Ramstead et al., 2018, Kirchhoff et al., 2018, Karl, 2012, Da Costa et al., 2023, Pezzulo et al., 2024]. This modeling approach has been used for various types of systems, from neural processes and cognitive functions to broader ecological and social dynamics [Friston et al., 2010, Da Costa et al., 2024, Solymosi and Schulkin, 2024, Albarracin et al., 2024a, Matsumura et al., 2023, Montgomery and Hipólito, 2023, Ramstead et al., 2020, Pezzulo et al., 2024].

While it is a widely held assumption that all systems will invariably minimize free energy (FE), this is not always a simple linear process. To understand this, we have to consider the system's goals and constraints. These goals can sometimes result in behaviors that do not align perfectly with immediate free energy minimization. This is partly what can make a system, given a specific scale of measurement, somewhat unsustainable. Not all systems are capable of effectively minimizing free energy. They may indeed have constraints in their structure or function. Think of certain pathological conditions impeding a system's ability to minimize free energy efficiently. These conditions can lead to maladaptive behaviors or states that deviate significantly from what would be predicted by the FEP. For example, constraints in structure or function: in neurological disorders such as schizophrenia, the brain's ability to minimize free energy can be impaired. As it distorts connectivity, it may also alter perceptions and thoughts - no longer fully related to the external world. Someone with Schizophrenia can struggle to reduce uncertainty about its environment, resulting in maladaptive behaviors [Friston et al., 2016, Harikumar et al., 2023, Zarghami et al., 2023].

Free energy minimization can also be influenced by external perturbations and environmental factors. The very nature of the environment is unpredictable dynamics, which temporarily disrupt a system's meta-stable states. The system transiently increases free energy as it adapts to new conditions. Albarracin et al. (2024) explore how systems must deal with external shocks and stresses (perturbations) to maintain sustainability, resilience and well-being. They suggest that resilience means absorbing shocks and stresses from the environment, while sustainability requires enduring capacity to stay resilient, but without causing a loss of resilience of the environment super-system. In this paradigm, external perturbations are central to developing better strategies to maintain well-being across system strata. Since these perturbations can be unpredictable, the temporality of strategies can change. Long-term strategies can weather slight increases in free energy temporarily to achieve more stable and favorable conditions in the future. This is the case we will be testing and presenting in this paper: long-term strategies involving temporary increases in free energy. When an agent learns that it does not have to satisfy its greed immediately, even if it is very hungry, because the aim is to maintain a balance between itself and the environment (such as a room with food) over time, the agent can resist the urge for immediate gratification and managing its resources judiciously. And thus, the agent can endure

short-term discomfort (increased free energy) to ensure long-term stability and sustainability.

Modeling sustainability through active inference allows us to model dynamically changing environments. Active inference is a probabilistic framework and thus allows agents to adapt their behaviors in response to fluctuations in resource availability. It is particularly relevant for studies on sustainability. For instance, in real-world scenarios such as climate change and resource depletion, the ability of agents to adjust their strategies based on anticipated future states is the critical element we need to model. Active inference accounts for the immediate needs of a system but also incorporates long-term goals that promote resilience and sustainability. In this study, we emphasize the importance of such adaptive strategies by illustrating how agents can navigate the challenges of resource management in environments that are subject to change.

To do so, we test out two cases, detailed in the Methods section. Case 1 acts as a baseline scenario, and involves a static environment where the agent decides whether to eat food or not. In Case 2, the environment is dynamic, and the agent must learn to moderate its consumption behavior over time. Food increases when the agent does not eat, introducing a dynamic aspect to the environment. This study is important for two reasons. First, we must understand adaptive strategies to properly predict when systems will achieve long-term stability and sustainability. It will help us predict when systems choose to balance short-term needs with long-term goals. But this will also help us identify potential vulnerabilities, such that we can identify areas where intervention may be needed to prevent collapse or dysfunction. The FEP dictates that behavior should align with minimizing free energy. But we have to better understand the variability in paths where this principle isn't consistently upheld at a given scale of measurement.

2 Methods

2.1 Case 1: Static environment

We build this simulation using the PyMDP package, by Conor Heins and colleagues [Heins et al., 2022]. In Case 1, we consider a static environment where the agent's goal is to maintain satiety by deciding whether to eat the available food. The generative model for Case 1 (detailed in Table 1, and visualized in figure 1 and 2) includes hidden states for food availability and agent satiety, observations that directly correspond to these hidden states, and actions to eat or not eat. The Likelihood Matrix ($\mathbf{A}$) assumes an identity mapping between hidden states and observations, meaning that the agent directly observes the true states of food availability and its own satiety. Mathematically, this is represented as $P(o_t \mid s_t, \mathbf{A}) = \text{Cat}(\mathbf{A})$, where $\mathbf{A}$ is an identity matrix. This means the probability of an observation given a state is 1 if they correspond and 0 otherwise:

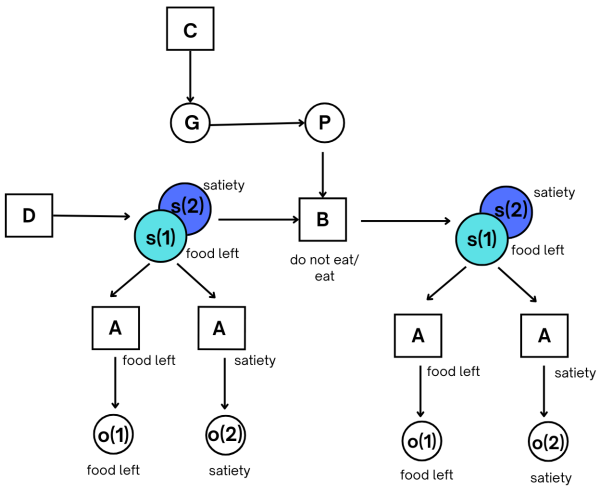


Fig. 1: The agent's generative model encodes beliefs about the causal structure of the environment and how its actions affect the state of the world. The true state of the environment is represented by two hidden state factors - the availability of food (s1) and the agent's satiety (s2). The prior preference C matrix specifies the agent's innate drives or goals, in this case a strong preference for being satiated. The starting conditions are specified by the initial state distribution, D. Here, food is initially present but the agent is not satiated. The agent has two observation modalities - the presence of food (o1) and its own satiety level (o2). The agent can select between two actions at each time step - "eat" or "do not eat". We have two hidden state factors: food left and satiety. For the "do not eat" action, for Case 1 the B matrix is an identity matrix, as this action does not change the state, while for Case 2 it changes, since not eating leads to an increase in available food. When the agent chooses the "eat" action, if food is present, the states will transition by reducing "food left" by 1 (down to a minimum of 0) state and by increasing "satiety" by 1 (to a maximum of 2).

$$P(o_t = i \mid s_t = j) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

The agent performs variational inference to optimize an approximate posterior $Q(s_t)$ over hidden states at each timestep, using the expected log likelihood of observations $\mathbb{E}[\log P(o_t \mid s_t)] = Q(s_t)^T \log \mathbf{A}$.

The Transition Matrix (B) specifies that the "eat" action leads to satiety when food is present, while food remains constantly available regardless of the agent's actions. The "don't eat" action leads to hunger.The transition likelihood B is represented as a set of matrices $B[f]$, one for each hidden state factor f, with dimensions $S_f \times S_f \times U_f$, where S_f is the number of levels for factor f and U_f is the number of control states or actions for that factor.

The entry $B[f][i, j, k]$ represents the probability of transitioning from state j to state i for factor f, given action k: $P(s^f_{t+1} = i \mid s^f_t = j, u^f_t = k)$. In this case, the "eat" action ($k = 0$) would have a high probability of leading to the "satiated" state (i) when the current state is "food available" (j), while the "don't eat" action ($k = 1$) would likely lead to the "hungry" state.

The transition matrices $B[f]$ are assumed to be conditionally independent across factors, meaning that the next state of factor f only depends on the current state and action for that factor, and not on the states of other factors: $P(s^f_{t+1} \mid s^f_t, u^f_t) = P(s^f_{t+1} \mid s_t, u_t)$. This simplifies the computation of the joint transition probability.

The Preference Vector (C) encodes a strong preference to observe satiation and food present. The agent's goals and preferences are represented as a prior distribution over observations, $P(o_{1:T})$. The C vector encodes these preferences as a categorical distribution, where higher values correspond to preferred observations. The agent aims to maximize the probability of sampling these preferred observations.

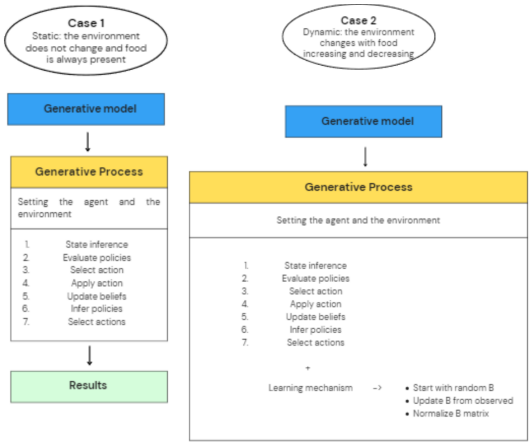


Fig. 2: Flowchart summarising case 1 and case 2 experimental setup.

The Initial State Distribution (D) is not specified, so that it is a uniform distribution where each state has an equal probability of being the initial state. In the PyMDP framework, the initial state distribution is represented as a categorical distribution over hidden states at the first timestep, $P(s_1 \mid D) = \text{Cat}(D)$. If not specified, it defaults to a uniform distribution, assigning equal probability to all possible initial states.

During the generative process, the agent interacts with the environment, and its actions affect the state transitions according to the generative process. If the

agent chooses not to eat, the state of the environment remains unchanged. If the agent chooses to eat and food is present, the agent becomes satiated, but food remains available due to the static nature of the environment. We then instantiate the simulation loop. First, the agent performs state inference based on the current observation, evaluating policies to maximize expected free energy, and selecting an action that minimizes free energy and aligns with its preferences. The selected action is applied to the environment, resulting in state transitions and new observations, and the loop continues with the agent updating its beliefs, inferring policies, and selecting actions to achieve its goals. In the extension to Case 1, we introduce variations to test the agent's adaptability. In Case 1.1, we set incorrect A and B matrices, introducing flawed perceptions and beliefs about state transitions. This extension is intended for us to validate that the behavior of the agent is in fact predicated on its appropriate appraisal of the environment.

Component	**Values**	**Description**
Hidden States	Food availability: present (1), absent (0)	Represents the true state of food availability in the environment
	Agent satiety: hungry (0), satiated (1)	Represents the true state of the agent's satiety
Observations	Observed food availability	Corresponds directly to the food availability state
	Agent's perceived satiety	Corresponds directly to the agent satiety state
Actions	Eat (1), Don't Eat (0)	The actions available to the agent
Likelihood Matrix (A)	Identity mapping	Assumes the agent directly observes the true states
Transition Matrix (B)	"Eat" (1) leads to satiety (1) when food is present (1)	Specifies the state transitions based on the agent's actions
	"Don't Eat" (0) leads to hunger (0)	
Preference Vector (C)	Strong preference for satiated (1) and food present (1)	Encodes the agent's goals and drives its behavior
Initial State Distribution (D)	Uniform Distribution	Sets the starting conditions for the simulation

Table 1: Components of the generative model for Case 1

2.2 Case 2: Dynamic Environment

In Case 2 (laid out in figure 2 (right) and table 2), we extend our model to a dynamic environment where the agent's actions have consequences on the availability of food resources over time. The goal is to study how the agent

adopts a sustainable behavior, balancing its immediate need for satiety with the long-term availability of food. The generative model for Case 2 (Figure 1) builds upon the previous model by introducing more granularity in the states and observations, allowing for a wider range of behaviors and interactions between the agent and the environment. Both the observations and hidden states are expanded to have three levels each: food left (0: none, 1: some, 2: abundant) and satiety (0: not satiated, 1: somewhat satiated, 2: fully satiated). In this model, we assume that the agent directly observes the true environmental states with some variations across different levels of food availability and satiety. In this dynamic environment, the transitions depend on both the current state and the action taken by the agent. If the agent does not eat, food availability increases over time, while if the agent eats, food availability decreases or remains depleted. For the satiety state, if the agent does not eat, satiety decreases over time, while if the agent eats, satiety increases. In Case 2, the preferences are designed to balance between maintaining satiety and ensuring a sustainable food supply, encouraging the agent to maximize its satiety while also considering the long-term availability of food resources. Specifically, the agent has a strong preference for being satiated, while flat preference over food left. The agent interacts with the dynamic environment over multiple time steps, updating its beliefs and actions based on the observed states and the changing dynamics of the environment, and it learns to not eat even if it is not fully satiated. The agent is initialized with the generative model specified in Case 2, using an extended policy length to plan multiple time steps ahead and anticipate future consequences. In the simulation loop, the environment starts with food fully available and the agent being half satisfied. The agent can plan over multiple time steps (policy length of 3), and thus has the opportunity to balance immediate consumption with long-term sustainability. The extended policy length allows the agent to anticipate future states and avoid greedy behavior that could lead to starvation. The agent's policies are restricted to ensure consistent and sustainable actions across all time steps for both observation modalities.

In Case 2.1, we extend the dynamic environment setup from Case 2 by introducing a learning mechanism for the agent. The key change is that the agent now starts with a random B matrix and updates it based on its experiences in the environment. The agent starts with a random B matrix instead of a predefined one, which will be updated as the agent interacts with the environment. The B matrix is initially random, and the agent updates this matrix based on observed transitions between states. The A matrix remains the same as in Case 2, mapping the hidden states to observations, and the C vector, representing the agent's preferences over observations, remains unchanged from Case 2. To learn, the agent starts with a randomly initialized B matrix, which does not initially capture the correct state transitions. The random initialization is done using a Dirichlet distribution to ensure valid probability values. At each time step, the agent receives observations, infers states, infers policies, and samples actions, similar to Case 2. After executing an action and receiving the next observation, the agent updates its B matrix. The agent notes the transition from the previous

Component	Values	Description
Hidden States	Food left: none (0), some (1), abundant (2)	Represents the true state of food availability in the environment
	Agent satiety: not satiated (0), somewhat satiated (1), fully satiated (2)	Represents the true state of the agent's satiety
Observations	Observed food availability	Corresponds to the food availability state with some variability
	Agent's perceived satiety	Corresponds to the agent satiety state with some variability
Actions	Eat (1), Don't Eat (0)	The actions available to the agent
Likelihood Matrix (A)	High probability of correct observations, lower for adjacent states	Defines the probability of observations given the true hidden states
Transition Matrix (B)	"Eat" (1): food left decreases, satiety increases	Specifies the state transitions based on the agent's actions and current state
	"Don't Eat" (0): food left increases, satiety decreases	
Preference Vector (C)	Strong preference for satiety. Balances maintaining satiety and sustainable food supply	Encodes the agent's goals and drives its behavior
Initial State Distribution (D)	Uniform Distribution	Sets the starting conditions for the simulation
Policy Length	3 time steps	Allows the agent to plan ahead and consider long-term effects

Table 2: Components of the generative model for Case 2

state to the current state given the action taken. The B matrix is updated using a learning rate to adjust the probabilities of the observed transitions. For states that depend on a single factor, the transition probability is updated directly by increasing the probability of the observed transition by the learning rate. For states that depend on two factors (e.g., satiety depends on both food left and previous satiety), the transition probabilities are updated based on the dependencies specified. After updating, the B matrix is normalized to ensure that the probabilities sum to 1, maintaining a valid probability distribution. We also introduce several extended case variations for Case 2 (with and without learning) to explore the agent's behavior and performance under different conditions. We test the agent's robustness by initializing the B matrix with incorrect values set to very high (1) or very low (0). The agent's performance is expected to degrade,

demonstrating the importance of accurate transition models, and avoiding inertia. We examine the agent's behavior when it has different prior preferences by modifying the C vector to represent a strong preference for food being present. The agent prioritizes actions that ensure food is present, potentially at the expense of satiety. We set a low preference on food (0) as well, and, as we expected, this helps the agents to learn the correct behaviour, since here it has a specific preference over avoiding no food left. In main case 2, however, he learnt by himself to save food, even though he was not meant to have a specific preference over a specific value for food left. We test the agent in an environment where food increases at a slower rate (0.5 units per step, compared to 1 unit per step previously) when not eating and decreases at a faster rate (1 unit per step) when eating. Satiety decreases faster when not eating (0.2 units per step, compared to 1 previously) and increases at a different rate when eating (0.8 units per step, compared to 1 previously). The agent needs to adapt its strategy to account for these specific changes in the environment dynamics. Its performance may be lower compared to Case 2 due to the increased difficulty in balancing food and satiety levels, as the food depletes more quickly when eating and satiety decreases more rapidly when not eating. Moreover, we simulate seasonal changing dynamics to test the agent's ability to retrain and to adapt to changes in the environment's rules after the initial training and adaptation depending on the different learning rates. So we introduce additional parameters for summer and winter, and update the environment's growth and depletion rates based on the current season. Additionally, we create a time-based switch in the simulation that changes the environment from summer to winter and vice versa after a certain number of time steps. We finally assess the impact of planning horizon on the agent's performance by comparing agents with different planning horizons (1 time step vs. 3 time steps). Agents with a longer planning horizon are expected to perform better, as they can anticipate future states more effectively and make decisions that lead to more sustainable resource management.

3 Results

In Case 1, the agent is in a static environment where food is always available, and its task is to maintain satiety by deciding whether to eat. The agent consistently chooses to eat at every time step, reflecting its understanding that food is always available and that eating maximizes its satiety (Figure 4, left, first row). Food availability remains constant throughout the simulation, as expected in a static environment where food does not deplete (Figure 4, left, second row). The agent's satiety increases as it eats and remains at a high level, indicating successful learning and adaptation to maintain its internal state optimally (Figure 4, left, third row). This setup demonstrates the agent's ability to perform optimally in an environment with constant resources. In Case 1.1, we introduce errors in the A and B matrices to test the agent's resilience and adaptability when its internal model does not accurately represent the environment. The agent's actions show a more erratic pattern, reflecting confusion or uncertainty due to

the incorrect matrices. Despite the confused matrices, food availability remains constant as in the standard case (Figure 4, right, second row). The agent's satiety fluctuates more compared to the standard case, indicating that the agent's ability to maintain a consistent internal state is impaired by the incorrect perception and planning models. This case shows how deviations from accurate environmental models can affect an agent's behavior and performance, leading to less optimal decisions. With Case 1 results (summarised in Table 3), we have shown that the agent has a degree of validity, and that it does in fact show how the agent reacts to model fitness, and chooses the best actions relative to its own survival.

Parameter	**Main Case**	**Incorrect A and B matrices**
Agent actions	Consistently chooses to eat at every time step	More erratic actions
Food availability	Remains constant throughout the simulation	Constant
Agent satiety	Increases as it eats and remains at a high level	Fluctuating

Table 3: Results for Case 1

In Case 2, the environment is dynamic, with food depleting when eaten and replenishing if not consumed. The agent must balance its eating behavior to avoid starvation and resource depletion. It is equipped with a strong preference over satiety = 2, and flat preference over food left (Appendix, figure 5). Flat preference where set to test the agent's ability to learn to conserve food and survive through time, although he was not explicitly told to do so. Indeed, the agent should not care about the availability of resources, but only about its satiety. Over multiple runs with a policy length of 3 time steps, the agent tends to avoid eating, leading him to die of starvation (Figure 4 top left), or eats too much, leading him to death as well (Figure 4 top right) as his food gets depleted. In Case 2 with learning, the agent starts with a randomly initialized B matrix and updates it through interactions with the dynamic environment. The agent's actions fluctuate regularly between "Eat" and "Do Not Eat," suggesting that it has learned a strategy to balance its actions, so that it manages to survive the whole time of the run and keeps satiety between 0 and 1 (Figure 4 bottom left). The survival time plot for each run shows that the agent consistently survives for the maximum number of time steps after the initial learning phase, indicating that it quickly learns an effective strategy to avoid starvation and maintain survival (Figure 4 bottom right).

Compared to Case 2 without learning, the case with learning shows the agent's ability to learn from its interactions with the environment and develop more effective strategies for survival and resource management. The extended Case 2 variations explore the agent's behavior and performance under different

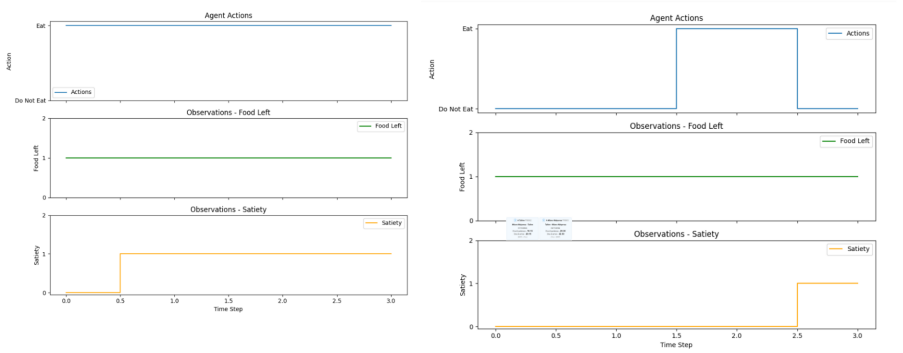


Fig. 2: Three plots on the left: Case 1. It is shown the expected behavior. At time step 0, food is available (food left = 0), and the satiety level is low (satiety = 0). Due to the agent's strong preference for high level of satiety, it keeps eating at subsequent time steps and the satiety increases. Since the environment is static, the food is always present.

Three plots on the right: Case 1.1 - where the agent is given incorrect A and B matrices, introducing errors in its perception and beliefs about state transitions. The top plot shows the agent's actions over time. The pattern is more erratic compared to the standard Case 1, as the agent is confused. The middle plot shows the food left observations. Food availability remains constant at 1 throughout the simulation since the environment is static. The bottom plot shows the agent's satiety over time. Satiety level fluctuates more. This indicates that the agent's ability to maintain a stable, high satiety state is impaired by the incorrect perception and planning models.

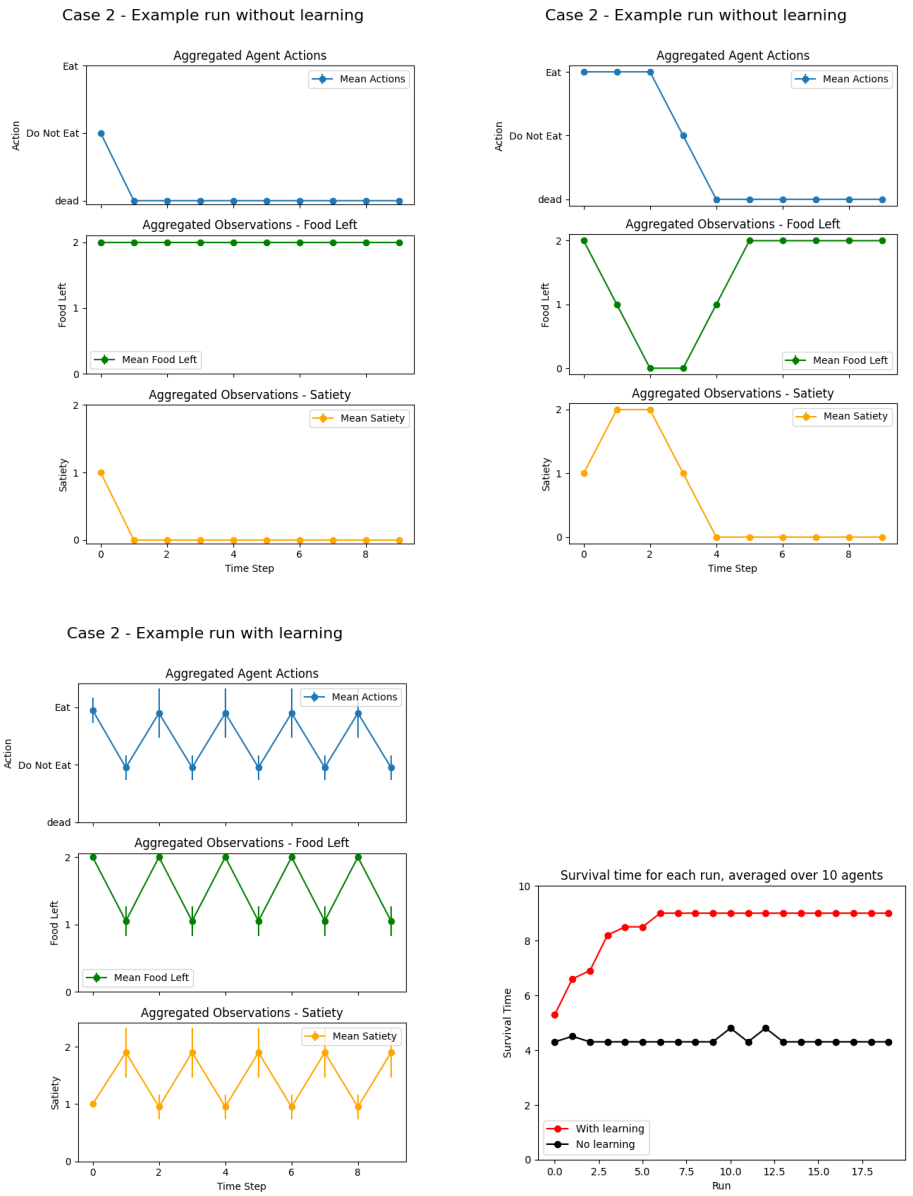


Fig. 4: Case 2 - Dynamic Environment. Without learning, the agent either does not eat (as shown in the three top plots on the left) or eats too much and therefore allows food in the environment to go to 0 (as shown in the three top plots on the right). As a result, the agent dies.

The three bottom plots on the left show an example run with learning on and policy length = 3. With this depth of policy, the agent is able to plan further in time, and with learning it manages to survive for the whole length of the run. Over 10 time steps the agent is able to plan its behaviour so that it never reaches satiety = 0, and always has food left.

Last bottom right plot shows the survival time of agents with both learning and no learning when the agent starts with a random B matrix. The survival time plot for each run averaged over 10 agents, shows that the agents can quickly learn to survive by acting in a sustainable way.

conditions, such as incorrect transition models, altered prior preferences, and varying planning horizons. With an incorrect B matrix, where the values are set to extremes from the start (1 and 0, rather than lower probabilities), the agent consistently chooses to eat in every time step, leading to suboptimal behavior and eventual starvation (Appendix, figure 6). Under certain conditions, the agent was unable to learn, being stuck in the inertia of its transitions. But overall, with learning enabled, the agent was able to pull itself out of high values and was able to survive - highlighting the value of plasticity to get out of bad bootstraps.

In the case of strong preferences on both states (high satiety and high food left - without learning), the agent initially chooses to eat but then stops eating as food becomes scarce, demonstrating the influence of strong preferences on the agent's actions (Appendix, figure 7, left). This leads the agent to die over most of its runs quite quickly. With learning enabled (Appendix, figure 7, right), the agent is able to balance its actions again and can survive longer, balancing its preferences and the environmental demands. By setting low preference on no food availability the agent's behavior is slightly the same.

When the environment rate of change changes (Appendix, figure 8), the agent's performance declines compared to the previous case, but learning still provides a significant advantage. With learning enabled, the agent adapts its strategy - eating less often to conserve food, maintaining higher average food levels, and sustaining satiety more effectively. This allows the agent to consistently survive the full run timesteps when learning, while it only survives around 3 timesteps without learning. Although the tougher environment dynamics make it more challenging, as the agent must plan in a different way and possibly over longer timescales, the agent demonstrates an impressive ability to adjust its policy through learning to match the new rate of change in food and satiety levels. Learning is critical for the agent to find the right balance and survive in this more complex scenario. If we simulate seasonal dynamics changes as shifting from summer to winter, the agent is able to learn the first season, but then it dies when it gets to the second season, because it does not have enough time to learn the new season (Appendix, figure 9). In this

Parameter	**Main Case**
Agent actions	Initially fluctuates between "eat" and "do not eat". Balances actions with learning
Food availability	Depletes when eaten, replenishes if not consumed. Agent balances food levels with learning
Agent satiety	Initially balances satiety and survival. Stabilizes with learning
Survival time	Agent consistently survives the maximum number of time steps after the initial learning phase

Table 4: Results for Case 2

case, it seems that one model is not sufficient for two different seasons, because it is difficult for the agent to learn, unlearn and relearn two different patterns. The last extension to case 2 gives a policy length of 1. Here the agent does very poorly without learning, dying basically after the first step. With learning, it takes the agent a little bit of time to learn how to survive, but it eventually does. Its actions are a little more erratic, but it does find a short term strategy (Appendix, figure 10). However, even with this short term strategy, it is unable to survive for very long, truly highlighting the need to focus on longer term strategies. Case 2 results are summarised in Tables 4 and 5.

4 Discussion

Our sustainable agent demonstrates how active inference can give rise to sustainable resource management strategies at the level of an individual agent. The agent's behavior emerges from the interaction between its model of the world, prior preferences, and the environmental dynamics. It seeks to optimize for immediate needs (e.g., hunger) and long-term outcomes (e.g., consistent food availability), learning to balance consumption and resource replenishment to promote sustainability. Our findings align with our previous formalization of sustainability, resilience, and well-being within the active inference framework.

In Case 1, the static environment allowed the agent to exhibit inertia, maintaining a consistent consumption pattern without considering long-term resource availability. While this behavior was adequate for the given context, it lacked the flexibility needed for sustainable outcomes in more dynamic environments. Case 2 introduced environmental variability, requiring the agent to demonstrate elasticity and plasticity. The agent's ability to adapt its eating habits in response to changing food availability exemplifies elasticity, as it temporarily endures increases in free energy (i.e., hunger) to ensure long-term stability. The agent's capacity to learn and update its model of the world based on new information reflects plasticity, enhancing its resilience in the face of environmental shifts. The agent's adaptive behavior in Case 2 reflects resilience, as it adjusts its actions to maintain well-being under changing resource availability. The dynamic coupling between agent and environment in the study of sustainable resource management was critical, even at the level of a single agent. In the extended cases, we can see the issues with inertia, and the possibility for even adaptive agents to get stuck in difficult policies. Case 2 extensions about different prior preferences show that an interesting area of research is where do prior preferences come from: we would expect agents to evolve to have a strong prior preference not to die. However it seems unlikely that an agent will evolve to have a preference for how much food it leaves in the environment, as is this is very environment dependent - it would be wasteful to have a prior preference to not use food which is in the environment if the food was able to regrow quickly. The key point from our research is that, if an agent wants to survive, it will learn to adapt to its environment and act as sustainably as is demanded by that environment. i.e. acting sustainably

is not something we do just for the sake of it. We act sustainably because that is what we need in order to survive.

The agent's actions optimized its own well-being and contributed to the resilience of the environment by preventing complete resource depletion. This reciprocal relationship between the agent and its environment is a fundamental aspect of sustainability, as the generative models of different layers in a hierarchical system are inherently linked through niche construction Albarracin et al. [2024b]. However, the model's simplicity also reveals its limitations. The single-agent, single-resource setup does not capture the complex interdependencies and feedback loops present in real-world systems.

Our model is susceptible to several potential pitfalls that warrant consideration. Overfitting may occur if the agent becomes too specialized to the specific environmental dynamics presented in our simulations, potentially limiting its adaptability to novel situations. The impact of inaccurate priors could significantly affect the agent's initial behavior and learning trajectory. We thus have to carefully calibrating these priors in real-world applications. The artificial nature of some settings used in our simulations, such as the use of incorrect matrices, while useful for testing robustness, may not fully represent the challenges faced in more realistic scenarios.

Future research should explore multi-agent scenarios with competing interests and shared resources, as well as environments with multiple, interconnected resource types and more sophisticated replenishment dynamics. Moreover, a problem of active inference as we have implemented it is that it takes a long time to change a model when the environment changes, especially if the model has been learned over a long time and has a high degree of model certainty. And our multi-season experiments confirm this. Interesting future work would be to focus on learning different discrete models for different environments and seasons Collis et al. [2024], Friston et al. [2023b] Additionally, the model does not consider the possibility of permanent resource depletion, which would require conditioning the environment's survival on the maintenance of certain values. In the future, we need to incorporate this aspect to understand the long-term implications of resource management strategies. To further advance the application of active inference in sustainable resource management, future work should focus on integrating network theory and dynamical systems theory to model and quantify the interdependencies between resources and their impact on overall system sustainability.

We must contextualize our approach within the broader landscape of sustainable resource management modeling. Compared to traditional approaches such as model-based reinforcement learning or model predictive control, active inference has advantages in its inherent tendency towards homeostasis and its ability to balance short-term needs with long-term stability. But it may be less efficient in scenarios requiring rapid optimization or in environments with highly deterministic dynamics where simpler models might suffice. Future work should conduct comprehensive comparative analyses to better understand the strengths and limitations of active inference in various sustainability contexts, potentially

leading to hybrid approaches that leverage the strengths of multiple modeling paradigms.

Optimizing precision or learning rates could also help foster the elastic and plastic resilience necessary for long-term sustainability and abundance. We would need to explore this avenue further. Our paper presents a proof-of-concept model demonstrating how active inference can inform sustainable resource management at the individual level.

We consider the relationship between agent and environment to highlight the importance of resilience, adaptability, and long-term planning in achieving sustainable outcomes. While the model's simplicity limits its direct applicability to real-world systems, it provides a foundation for future research exploring the complex dynamics of sustainable resource management through the lens of active inference.

Bibliography

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

T. Parr and K. J. Friston. Generalised free energy and active inference. *Biological Cybernetics*, 113(5):495–513, 2019.

G. Stubbs and K. Friston. The police hunch: the bayesian brain, active inference, and the free energy principle in action. *Frontiers in Psychology*, 15:1368265, 2024.

K. Friston, T. FitzGerald, F. Rigoli, P. Schwartenbeck, and G. Pezzulo. Active inference: a process theory. *Neural Computation*, 29(1):1–49, 2017.

K. Friston, L. Da Costa, N. Sajid, C. Heins, K. Ueltzhöffer, G. A. Pavliotis, and T. Parr. The free energy principle made simpler but not too simple. *Physics Reports*, 1024:1–29, 2023a.

T. Parr, K. Friston, and G. Pezzulo. Generative models for sequential dynamics in active inference. *Cognitive Neurodynamics*, pages 1–14, 2023.

M. J. D. Ramstead, P. B. Badcock, and K. J. Friston. Answering schrödinger's question: A free-energy formulation. *Physics of Life Reviews*, 24:1–16, 2018.

M. Kirchhoff, T. Parr, E. Palacios, K. Friston, and J. Kiverstein. The markov blankets of life: autonomy, active inference and the free energy principle. *Journal of The Royal Society Interface*, 15(138):20170792, 2018.

F. Karl. A free energy principle for biological systems. *Entropy*, 14(11):2100–2121, 2012.

L. Da Costa, N. Sajid, T. Parr, K. Friston, and R. Smith. Reward maximization through discrete active inference. *Neural Computation*, 35(5):807–852, 2023.

G. Pezzulo, T. Parr, and K. Friston. Active inference as a theory of sentient behavior. *Biological Psychology*, page 108741, 2024.

K. J. Friston, J. Daunizeau, J. Kilner, and S. J. Kiebel. Action and behavior: a free-energy formulation. *Biological Cybernetics*, 102:227–260, 2010.

L. Da Costa, S. Tenka, D. Zhao, and N. Sajid. Active inference as a model of agency. *arXiv preprint arXiv:2401.12917*, 2024.

T. Solymosi and J. Schulkin. Creative resilience. flourishing and valuation through social allostasis and active inference. *European Journal of Pragmatism and American Philosophy*, 16(XVI-1), 2024.

M. Albarracin, G. Bouchard-Joly, Z. Sheikhbahaee, M. Miller, R. J. Pitliya, and P. Poirier. Feeling our place in the world: an active inference account of self-esteem. *Neuroscience of Consciousness*, 2024(1):niae007, 2024a.

T. Matsumura, K. Esaki, S. Yang, C. Yoshimura, and H. Mizuno. Active inference with empathy mechanism for socially behaved artificial agents in diverse situations. *Artificial Life*, pages 1–21, 2023.

C. Montgomery and I. Hipólito. Resurrecting gaia: harnessing the free energy principle to preserve life as we know it. *Frontiers in Psychology*, 14:1206963, 2023.

M. J. Ramstead, K. J. Friston, and I. Hipólito. Is the free-energy principle a formal theory of semantics? from variational density dynamics to neural and phenotypic representations. *Entropy*, 22(8):889, 2020.

K. Friston, H. R. Brown, J. Siemerkus, and K. E. Stephan. The dysconnection hypothesis (2016). *Schizophrenia Research*, 176(2-3):83–94, 2016.

A. Harikumar, K. P. Solovyeva, M. Misiura, A. Iraji, S. M. Plis, G. D. Pearlson, ..., and V. D. Calhoun. Revisiting functional dysconnectivity: A review of three model frameworks in schizophrenia. *Current Neurology and Neuroscience Reports*, 23(12):937–946, 2023.

T. S. Zarghami, P. Zeidman, A. Razi, F. Bahrami, and G. A. Hossein-Zadeh. Dysconnection and cognition in schizophrenia: A spectral dynamic causal modeling study. *Human Brain Mapping*, 44(7):2873–2896, 2023.

C. Heins, B. Millidge, D. Demekas, B. Klein, K. Friston, I. Couzin, and A. Tschantz. pymdp: A python library for active inference in discrete state spaces. *arXiv:2201.03904*, 2022.

M. Albarracin, M. Ramstead, R. J Pitliya, I. Hipolito, L. Da Costa, M. Raffa, A. Constant, and S. G. Manski. Sustainability under active inference. *Systems*, 12(5):163, 2024b.

P. Collis, R. Singh, P. Kinghorn, and C. Buckley. Learning in hybrid active inference model. *ArXiv preprint*, 2024.

K. Friston, A. Da Costa, L. Tschantz, A. Kiefer, T. Salvatori, V. Neacsu, M. Koudahl, Sajid N. Heins, C. and, D. Markovic, T. Parr, T. Verbelen, and C. Buckley. Supervised structure learning. *arXiv:2311.10300*, 2023b.

5 Appendix 1 - Figures

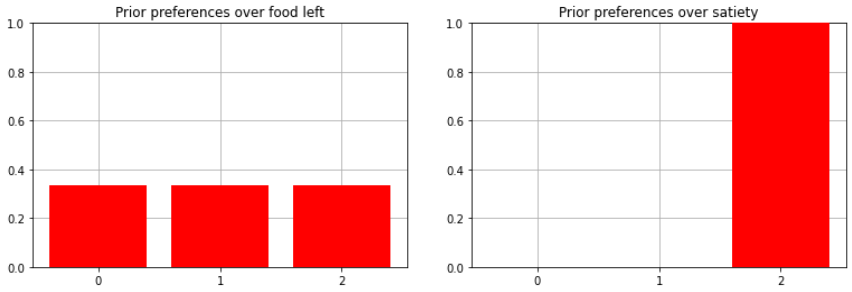


Fig. 5: Prior preference for Case 2. The agent's preferences are changed so that, unlike case 1 and 1.1 it no longer has a preference over food left. Its only non-uniform preference is to have a preference over satiety.

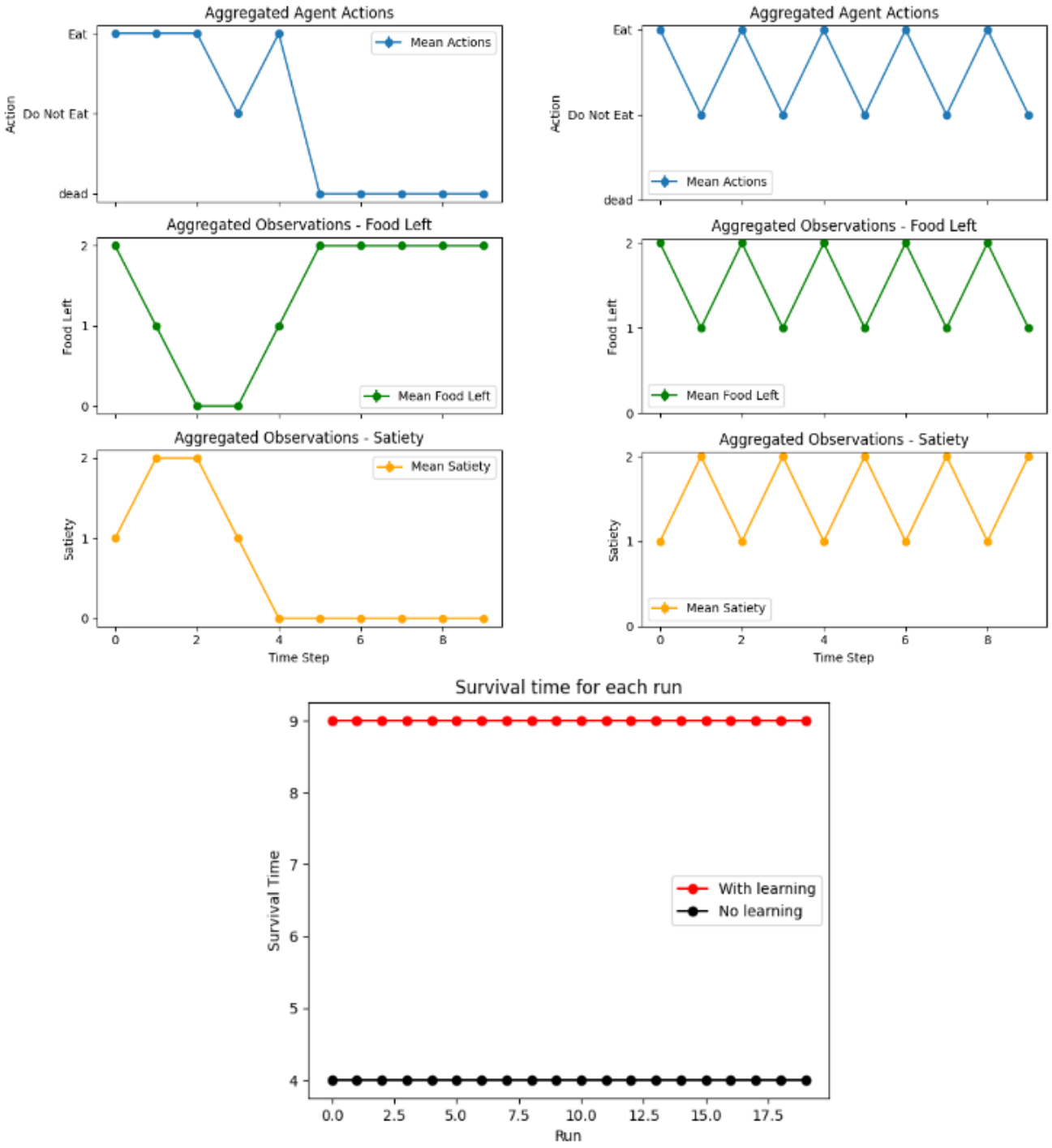


Fig. 6: Example run from Case 2 without learning enabled on the left and with learning enabled on the right, but starting with an extreme B matrix setting (probabilities set to 1 or 0, on the left three plots and middle three plots, and high but non-extreme values on the right). The agent dies quickly, just as the randomly set values of the B matrix in plot 6, and is able to learn on the right.

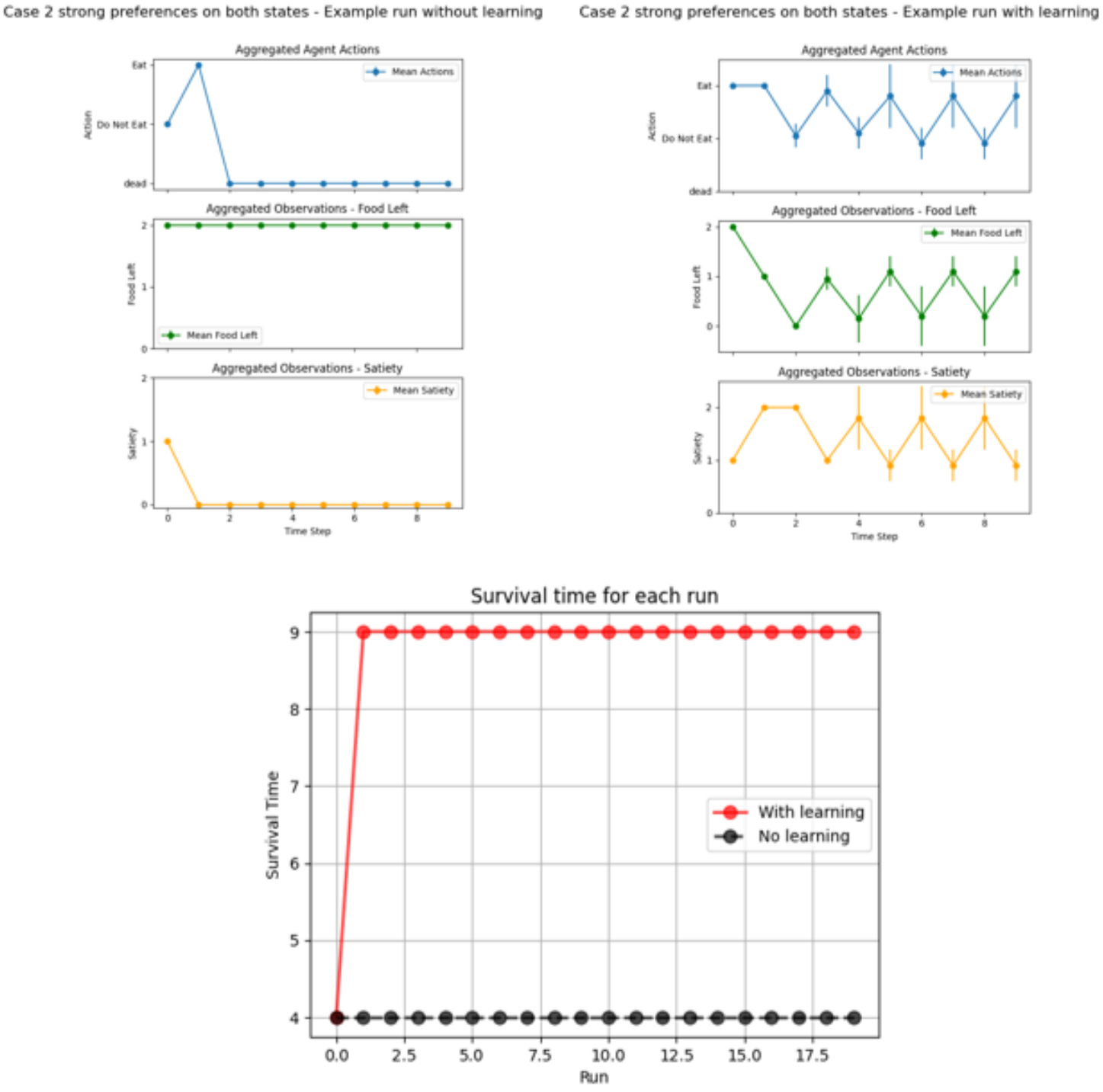


Fig. 7: Case 2 with strong prior preferences on both high satiety and high food left states. On the three top left plots, the agent has no learning, and on the top right, the agent has learning. On the bottom, we can see that survival time is vastly different with and without learning, as the preferences affect the behavior of the agent.

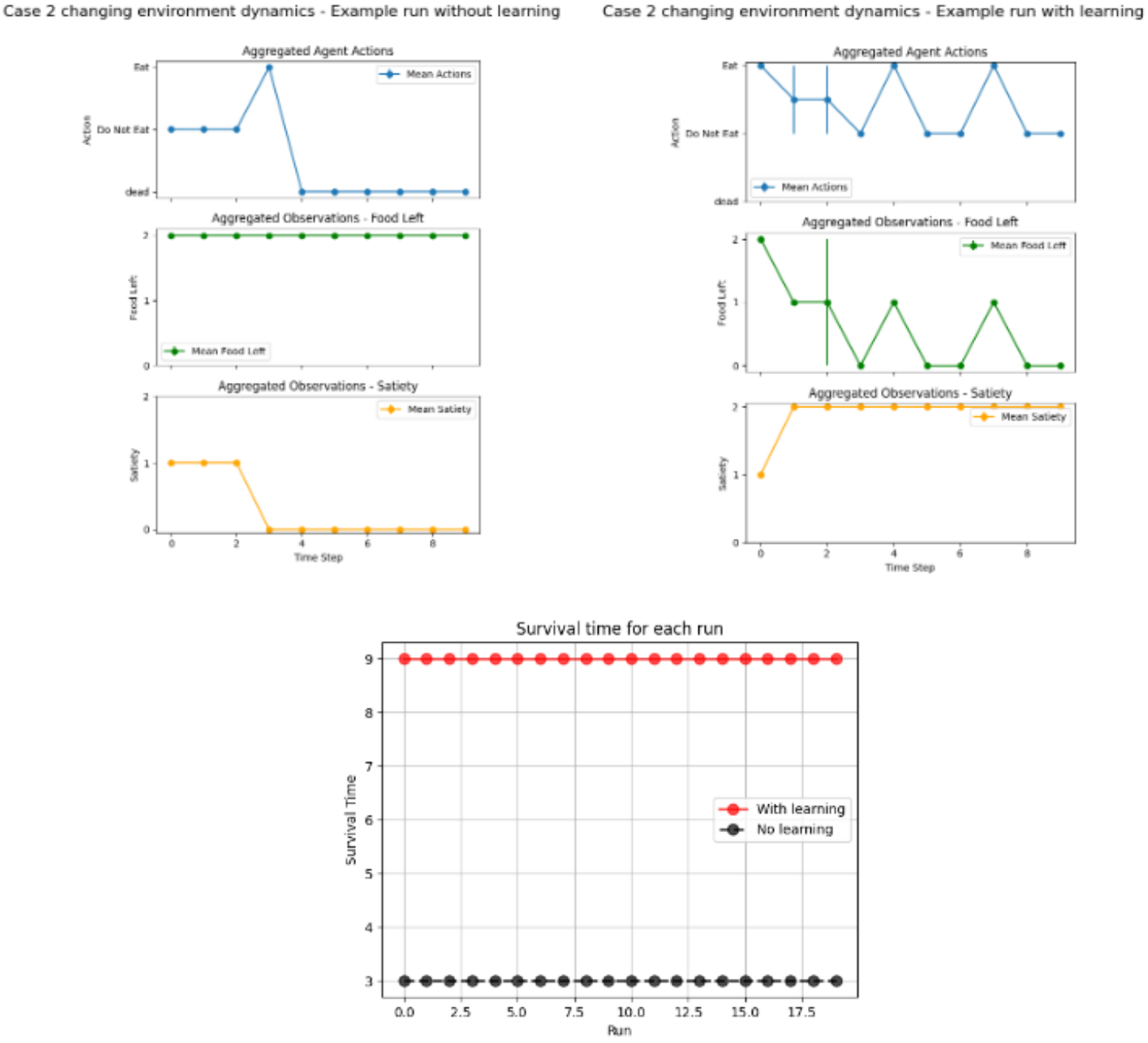
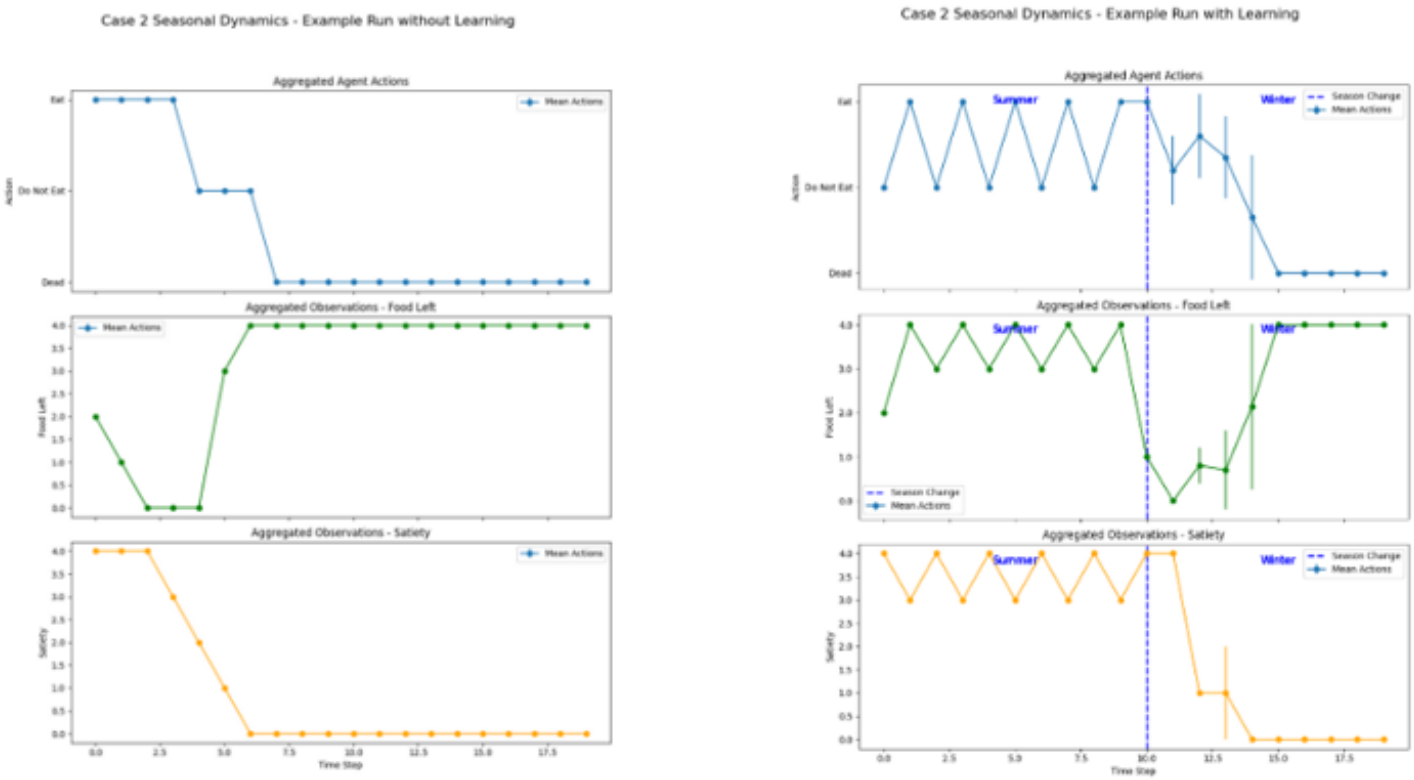


Fig. 8: Case 2 in a changing environment where food and satiety change at different time rates. The three top left plots show the results without learning off, and the three top right plots show the results with learning on. The bottom plot represents the comparison between the survival time over 10 time steps. Food increases at a slower rate (0.5 units per step) when not eating and decreases at a faster rate (1 unit per step) when eating. Satiety decreases faster when not eating (0.2 units per step) and increases at a different rate when eating (0.8 units per step).



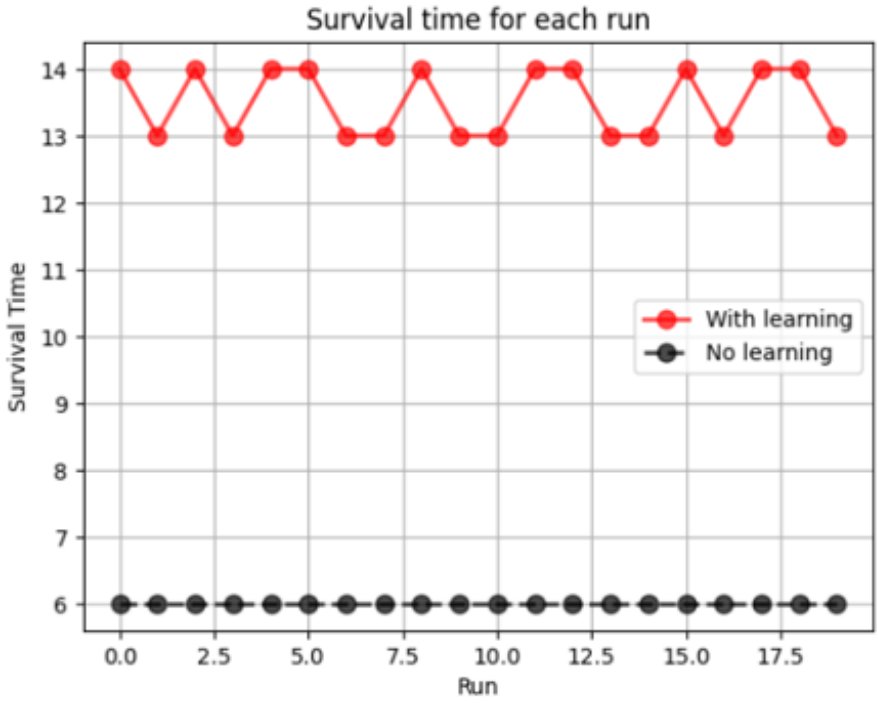


Fig. 9: Case 2 in a changing environment with seasonal simulation. The environment is built to simulate switching seasons summer/winter. With learning off the agent is not able to survive throughout the first season (top left plot and bottom plot). With learning on, it survives but then quickly dies throughout winter (top right plot).

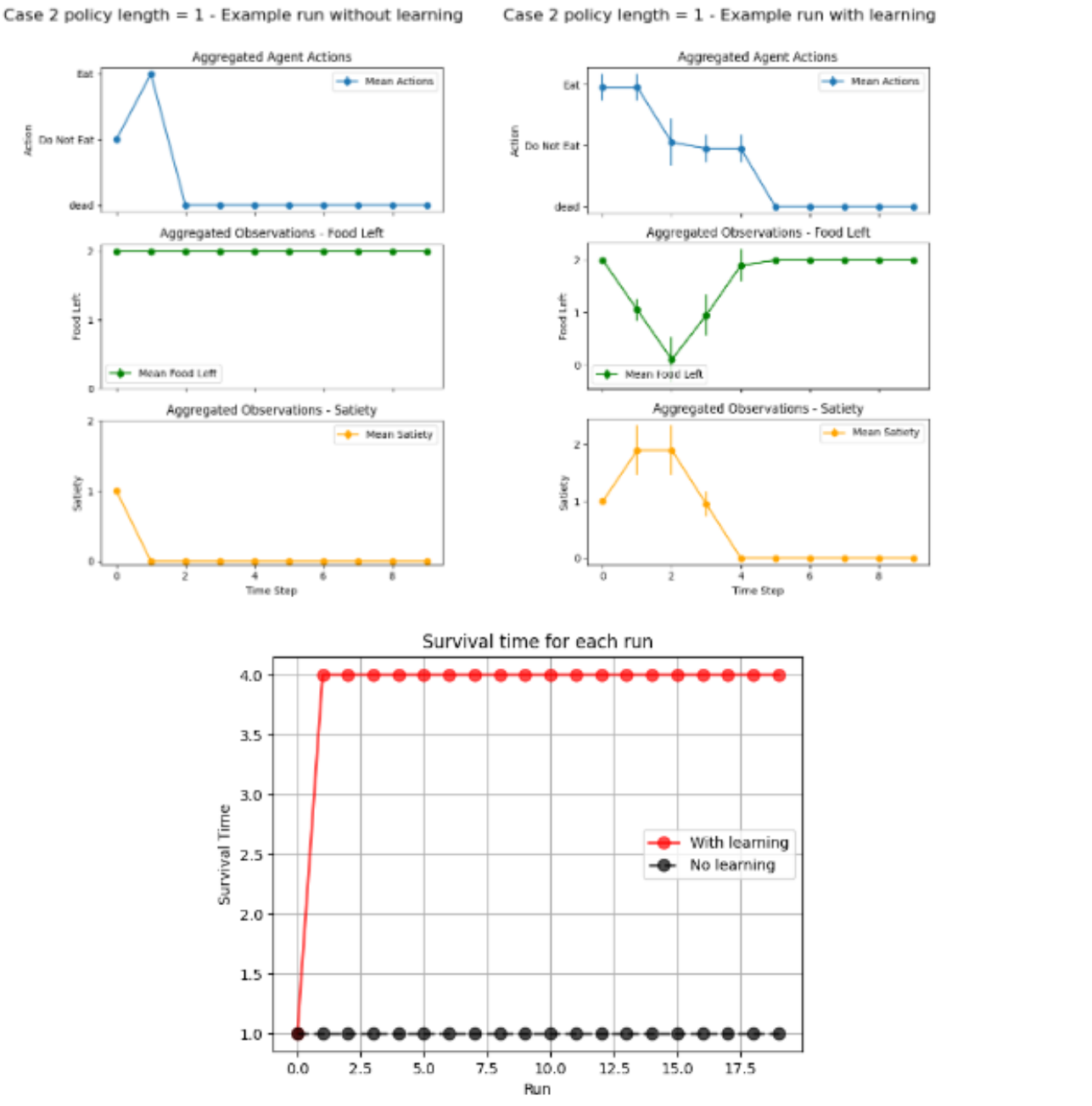


Fig. 10: Case 2 example runs runs with policy length = 1, left plot without learning, right plot with learning, and survival time on the bottom.

1. Incorrect B Matrix		
Initial B matrix setting	Extreme values (0 or 1)	
Agent actions	Consistently chooses to eat, leading to suboptimal behavior and eventual starvation	
Learning	Able to learn and survive by adjusting extreme values with learning	
2. Different preferences		
Preference	Strong preference for both high satiety and high food left	Low preference on no food left (0)
Agent actions	Initially chooses to eat, then stops as food becomes scarce, leading to quick death without learning	Quick death without learning
Learning	Balances actions and survives longer	Balances actions and survives longer
3. Different environment rate of change		
Food rate change	Increases at 0.5 units/step when not eating; decreases at 1 when eating	
Satiety rate change	Decreases at 0.2 unit/step when eating; increases at 0.8 when not eating	
Agent actions	Adaptive with learning; learns to conserve food and maintain satiety	
Survival time	Consistently survives full run time with learning; only around 1/3 time without learning	
3.1 Seasonal simulation		
	Summer	Winter
Food rate change	Increases at 3 when not eating; decreases at 1 when eating	Increases at 1 when not eating; decreases at 3 when eating
Satiety rate change	Increases at 3 when not eating; decreases at 1 when eating	Increases at 1 when eating; decreases at 3 when not eating
Agent actions	Adaptive with learning	
Survival time	Without learning, the agent dies quickly before season shifting. With learning manages to survive throughout summer but quickly dies during winter	
4. Different policy length		
Policy length	Time step = 3 vs Time step = 1	
Agent actions	Performs poorly with 1 time step without learning; more stable but same strategy with learning	
Survival time	Unable to survive long with 1 time step; longer survival with learning	

Table 5: Results for Case 2 extended variations

Message Passing-based Bayesian Control of a Cart-Pole System

Sepideh Adamiat[1], Wouter M. Kouw[1], Bart van Erp[1,2], and Bert de Vries[1,2]

[1] Electrical Engineering Department, TU Eindhoven, Netherlands
[2] Lazy Dynamics B.V., Eindhoven, Netherlands
s.adamiat@tue.nl

Abstract. We describe a Bayesian controller for a cart-pole system, a well-known benchmark in control theory. The cart-pole system is characterized by its nonlinear and underactuated nature, and we further complicate the scenario by (1) assuming that the controller lacks knowledge of sensor noise variance, and (2) imposing bounds on the control signal. Traditional control algorithms often struggle to adapt to uncertainties and constraints. However, the Bayesian framework, particularly the active inference framework, smoothly accommodates these complexities. In the proposed controller, the entire computational process consists of online Bayesian inference. This process is streamlined through a toolbox for fast message passing-based inference in factor graphs. We describe the mechanics of message passing in factor graphs, addressing challenges such as non-linear factors, bounded control, and real-time parameter tracking. The primary objective of this paper is to demonstrate that, with the advancement of the active inference framework and the effectiveness of automated inference toolboxes, Bayesian control emerges as an appealing option for application engineers.

Keywords: Active inference · Bayesian control· Factor graphs· Message passing · NUV priors · Policy estimation

1 Introduction

The cart-pole problem (also known as the inverted pendulum) comprises a pole that is attached at one end to a movable cart with an associated goal of balancing the pole at the upright position by controlling the horizontal movements of the cart.This is a highly non-linear and underactuated system [3] that is widely used as a benchmark to illustrate the effectiveness of controllers [21]. Despite the challenges, a standard cart-pole system can be successfully controlled by classical control algorithms such as Model Predictive Control (MPC)[15].

In the real world, control systems inevitably have to deal with uncertainties that result from model inaccuracies and simplifications such as un-modeled external influences or sensor imperfections. In this paper, we add some complexity

[3] A cart-pole system has two degrees of freedom, namely the pendulum angle and the linear cart position, and only one actuator, the horizontal force on the cart.

to the cart-pole control task by assuming that (1) the variance of the sensor noise is unknown, and (2) the control signal is bounded.

Active inference is a mathematical framework designed for understanding biological agents, specifically the human brain [6]. This framework is grounded in a generative model that drives states, controls, and planning based on the principle of free energy minimizing [4, 19]. Recently, many studies have used active inference agents in robotic tasks [13, 3]. In this paper, due to the highly nonlinear and underactuated behavior of the cart pole system, we are interested in active inference agents with continuous control and state spaces [6, 12, 13].

Current (non-Bayesian) control algorithms have difficulties quantifying or processing such uncertainties appropriately [20]. In contrast, Bayesian control through probabilistic inference in a generative model provides a principled way to keep account of uncertainties and constraints in the system. Unfortunately, when trying to realize a Bayesian controller, exact inference quickly becomes computationally intractable, even for relatively simple models. Numerical solutions such as (Monte Carlo) sampling-based inference are often too computationally intensive or too slow for the application at hand.

This paper presents an approach based on casting both the control and parameter tracking problems as online inference tasks on a generative model of the system. To combat the computational issues surrounding probabilistic inference, we realize the inference tasks by message passing (MP) on a Forney-style factor graph (FFG) representation of the probabilistic model. Efficient probabilistic inference in the model is realized using automatable MP procedures that leverage the conditional independencies in the model. MP-based inference has a long history for efficient inference in signal processing and control systems [14].

To keep the control signals within physical bounds, we use a Normal-with-Unknown-variance (NUV) distribution as a prior distribution for the control signals. The NUV prior is a distribution that originated in the sparse Bayesian learning literature [22]. Recently, NUV priors were introduced as a sub-model to enforce domain constraints [10]. This sub-model has been successfully used in various MPC applications to impose constraints on the state trajectories [10], as well as in multi-agent trajectory estimation to prevent collisions [2].

In section 2, we introduce the cart-pole system formally and specify the control problem. The subsequent sections include our contributions:

- In section 3, we specify the controller, which comprises a probabilistic generative model for sensory observations from its environment. Crucially, the model can be used to predict both veridical and desired future observations.
- In section 4, we rehearse factor graphs and various message passing methods that can be used to automate the control-by-inference process.
- Finally, in section 5, we evaluate the proposed Bayesian control method for the cart-pole system in a simulated environment.

In short, the novelty of this paper lies not in the introduction of any specific technique, but rather this paper aims to demonstrate at a systems engineering level how to realize a complex Bayesian control system. We bring together various methods to show that Bayesian control with both uncertainties and constraints

can be systematically realized through the specification of a biased generative model and a fully automatable inference process.

2 Problem Setting

2.1 The Cart-Pole System

In this paper, we simulate the cart-pole system as a state space model. In this model, the state variables are defined as a vector $z = [x, \theta, \dot{x}, \dot{\theta}]$, where $x \in \mathbb{R}$, $\theta \in \mathbb{R}$, $\dot{x} \in \mathbb{R}$, and $\dot{\theta} \in \mathbb{R}$ are the cart position, the pole angle, the cart velocity, and the angular velocity of the pole, respectively. Based on a Lagrangian mechanics approach, [21] derives the following equations of motion:

$$u = (m_c + m_p)\ddot{x} + m_p l\ddot{\theta}\cos\theta - m_p l\dot{\theta}^2 \sin\theta \tag{1a}$$

$$0 = m_p l\ddot{x}\cos\theta + m_p l^2\ddot{\theta} + m_p \mathrm{g} l \sin\theta\,, \tag{1b}$$

where m_c is the cart mass, m_p is the pendulum mass, g is the gravitational constant, and l is the pendulum length. These variables are assumed to be constant. Furthermore, $u \in \mathbb{R}$ is the horizontal force applied to the car, $\ddot{x} \in \mathbb{R}$ is the cart acceleration, and $\ddot{\theta} \in \mathbb{R}$ is the angular acceleration. For simulation purposes, based on Euler's method, we derive a discrete-time state space model with state variables $z_t = [x_t, \theta_t, v_t, \omega_t]$ where the x_t, θ_t, v_t and ω_t are the cart position, pole angle, cart velocity and the angular velocity of the pole at time t. This discrete-time state space model is then defined as

$$\underbrace{\begin{bmatrix} x_{t+1} \\ \theta_{t+1} \\ v_{t+1} \\ \omega_{t+1} \end{bmatrix}}_{z_{t+1}} = \overbrace{\underbrace{\begin{bmatrix} x_t \\ \theta_t \\ v_t \\ \omega_t \end{bmatrix}}_{z_t} + \begin{bmatrix} v_t \\ \omega_t \\ a_t \\ \alpha_t \end{bmatrix} \cdot \Delta t}^{g(z_t, u_t)} \tag{2}$$

where Δt is the interval between two time steps, and the a_t and α_t are the cart acceleration and the angular acceleration at time t. By using the equation of motion (1), the a_t and α_t at time t can be derived as

$$a_t = \frac{u_t + m_p \sin\theta_t (l\omega_t^2 + g\cos\theta_t)}{m_c + m_p \sin^2\theta_t} \tag{3a}$$

$$\alpha_t = \frac{\text{-}u_t \cos\theta_t - m_p l\omega_t^2 \cos\theta_t \sin\theta_t - (m_c + m_p)\mathrm{g}\sin\theta_t}{l(m_c + m_p \sin^2\theta_t)}. \tag{3b}$$

For more details, we refer to [21]. In the following, we also use the abbreviation $g(z_t, u_t)$ to indicate the right-hand side of (2).

The initial state of the cart-pole system is given by $z_0 = [0, -\pi, 0, 0]$, indicating that the pole is initially in a downward position and both the cart and the pole are at rest.

2.2 The Control Problem

We assume that the horizontal force signal u_t can be selected by a control agent. The agent interacts with its environment through a series of trials. The agent observes the cart-pole system's state through a sensor and executes actions u_t via an actuator that connects to the cart.

We assume that the agent partially observes its environment using a measurement matrix C with measurement noise v_t:

$$y_t = Cz_t + v_t\,, \tag{4}$$

with

$$C = \begin{bmatrix} 1\,0\,0\,0 \\ 0\,1\,0\,0 \end{bmatrix} \quad \text{and} \quad v_t \sim \mathcal{N}(0, R^*)\,, \tag{5}$$

where R^* is a fixed covariance matrix.

The goal of the agent is to guide the cart-pole system towards a desired goal state $z^* = [0, 0, 0, 0]$, where the pole is steady at the upright position. Given that direct control over the pole is not feasible (the controller can only apply a force to the cart), the control agent needs to *infer* a control sequence (policy) $u_1, u_2, u_3, \ldots$ to reach z^*.

We take a fully Bayesian inference approach to control [18]. In this framework, the controller is equipped with a probabilistic generative model for its observations. This model can be run forward in time to create predictions of future observations. As will be discussed in section 3.2, these predictions are biased toward desirable observations, and Bayes-optimal control signals can be obtained through online Bayesian inference.

3 The Probabilistic Controller Model

3.1 Specification of Generative (Veridical) Model

This subsection specifies the control agent for the simulated cart-pole system. The controller's model is a *generative* model, so we will specify a model with hidden dynamics that leads to probabilistic predictions of sensory inputs y_t.

We assume that the controller knows the dynamics of the cart-pole system, with uncertainty characterized by Gaussian process noise. Hence, the controller's state transition model is given by

$$p(s_{t+1} \mid s_t, u_t) = \mathcal{N}(s_{t+1} \mid g(s_t, u_t), Q)\,, \tag{6}$$

where s_t is controller's internal state, and Q is a fixed and known covariance matrix. In general, we run our simulation from an initial state $p(s_0) = \mathcal{N}(s_0 \,|\, z_0, Q)$. We also assume that the controller can predict its sensory observations with a "correct" observation model:

$$p(y_t \mid s_t, R_t) = \mathcal{N}(y_t \mid Cs_t, R_t)\,. \tag{7}$$

where C is defined as in (5) and R_t is a measurement noise covariance matrix at time step t. To complicate matters, we assume that the controller does not know the observation noise covariance matrix R_t, and therefore we will infer the appropriate value of R_t online, alongside inference for states s_t and controls u_t. The assumption is that the covariance matrix has a fixed unknown value, specified to be

$$p(R_t|R_{t-1}) = \delta(R_t - R_{t-1}) \tag{8a}$$

$$p(R_0) = \mathcal{W}^{-1}(R_0 \mid V_0, n_0), \tag{8b}$$

where $\mathcal{W}^{-1}$ is an inverse-Wishart distribution, which is a conjugate prior for the covariance matrix of a multivariate normal distribution.

To complete the generative model, we assume independent priors $p(u_t)$ over admissible actions at each time step. We will choose a Normal-with-Unknown Variance (NUV) prior, which effectively renders a "box constraint" of the form

$$p(u_t) \propto \exp\left\{-\gamma(|u_t - a| + |u_t - b|)\right\}, \tag{9}$$

where a and b are user-selectable control limits, γ is a parameter to control the softness of the constraint and $\propto$ means "is approximately proportional to". The exact form of the prior and subsequent inference procedure is discussed in section 4.5.

The set of equations (6), (7), (8) and (9) is called the agent's veridical generative model since it aims to predict the evolution of future sensory inputs according to the agent's knowledge about the environmental dynamics.

3.2 Specification of Target Model for Observations

The veridical model specification in section 3.1 can be used by the controller to predict future sensory inputs. In a Bayesian active inference framework, sensory targets are specified by extending the veridical model by *target* distributions [5]. These target distributions lead to goal-oriented behavior(such as guiding a cart-pole system to a target state) through Bayesian inference. We assume the following target distribution for observations:

$$p'(y_t) = \begin{cases} \mathcal{N}(y_t \mid Cz^*, 10^8 \cdot I) & \text{if } t \leq T \\ \mathcal{N}(y_t \mid Cz^*, 10^{-8} \cdot I) & \text{otherwise.} \end{cases} \tag{10}$$

The prime in $p'(\cdot)$ indicates that this distribution relates to desired or target beliefs, rather than veridical beliefs. Eq. (10) expresses that, for the first T time steps, the agent has essentially no preference for observations, but any time after T, the agent has a strong preference for receiving $y_t \approx Cz^*$.

The vague prior for $t \leq T$ allows the agent to infer actions that are most informative about the uncertainties in the model, such as the value of R_t. This is an explorative phase. For $t > T$, the tight target priors add an incentive to infer actions u_t that drive the cart-pole to its target state.

Taking the veridical and target beliefs together, the complete generative model for the controller can be represented as

$$p(y, s, u, R) \propto p(s_0)p(R_0) \cdot \prod_{t>0} p'(y_t)p(y_t \mid s_t, R_t)p(s_t \mid s_{t-1}, u_t)p(u_t)p(R_t|R_{t-1})\,. \tag{11}$$

Note that, due to the extension with the target prior $p'(y_t)$, the controller holds a *biased* model of the future! When unrolling this model into the future, this model predicts desired future observations, in the context of given assumptions about how the world "really" works (as specified by the veridical model). This approach aligns with the active inference framework that is claimed to describe a biologically plausible approach to control in living systems [5].

4 Inference

4.1 Inference is the Only Ongoing Process

Since the controller's generative model is biased toward predicting target observations, the actual process to be executed by the controller is just continual inference over all latent variables as new data y_t keeps streaming in through its sensory channels. This online inference process will update beliefs over its internal states s_t, the latent control variables u_t, and the latent covariance matrix R_t.

For a complex non-linear system with some non-conjugate distribution pairings, such as this dynamic Cart-pole controller, it is not possible to derive closed-form analytical Bayesian inference solutions, and sampling-based inference methods are usually too slow for real-time systems. Therefore, in this paper, we automate the online inference process through efficient message-passing-based inference on a factor graph representation of the controller's model.

Fortunately, we do not need to derive all messages from scratch. The open-source Julia package RxInfer supports fast message passing-based inference for a large range of models [1]. Next, we rehearse factor graphs and message passing-based inference.

4.2 Forney-style Factor Graphs and the Sum-Product Rule

A Forney-style Factor Graph (FFG) is a graphical representation of a factorized probabilistic model [14]. In an FFG, edges represent random variables and nodes represent factors, which are functions that specify the relationships between the variables. An edge connects to a node if and only if the variable on that edge is an argument of the node's function. Figure 1 shows the FFG for the probabilistic model defined in (11). In an FFG, each edge can maximally connect to two nodes. If a variable is an argument in more than two factors, we introduce a

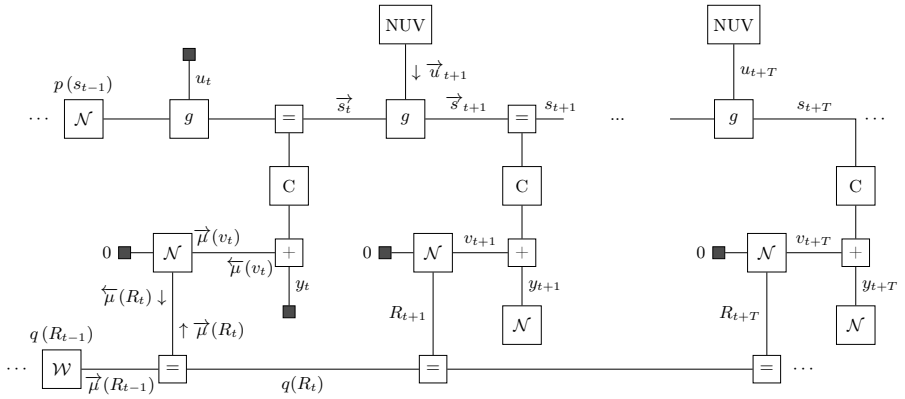


Fig. 1: A Forney-style factor graph representation of the probabilistic control model in (11). This snapshot illustrates the model at the time step t, with T future time horizon.

"branching" (also known as "equality") node that effectively copies the variable to an auxiliary variable with the same beliefs.

Aside from model visualization, FFGs support efficient message passing-based (MP) inference on the graph. MP-based inference is a highly efficient tool for performing probabilistic inference on sparsely connected generative models [14]. It scales well to large inference tasks and significantly speeds up Bayesian inference by effectively taking advantage of the distributive law ($ab+ac = a(b+c)$), which converts an (expensive) sum-of-products to a (cheaper) product-of-sums. We use $\overrightarrow{\mu}(\cdot)$ and $\overleftarrow{\mu}(\cdot)$ notations for the forward and backward messages respectively. Following the recipe above of moving factors over integrals (or summation signs), marginalization and Bayesian inference turn into a sequence of updating messages. These messages can be computed by the so-called sum-product rule [14].

In general, for any node $f(y, x_1, \ldots, x_n)$, the sum-product rule for an outgoing message over edge y is given by

$$\overrightarrow{\mu}(y) = \int \underbrace{\overrightarrow{\mu}(x_1)...\overrightarrow{\mu}(x_n)}_{\text{incoming messages}} \underbrace{f(y, x_1, ..., x_n)}_{\text{node function}} \mathrm{d}x_1...\mathrm{d}x_n \,. \tag{12}$$

Note that the MP algorithm minimizes the Bethe Free Energy (BFE)[23], which is known to lack epistemic (information-seeking) qualities. Consequently, agents using MP do not proactively seek informative states. Ongoing research seeks to address this limitation, for example, through the development of Constrained BFE (CBFE)[11]. The CBFE allows for inference that benefits from the MP algorithm's scaling advantages while retaining the epistemic qualities of expected free energy. Unfortunately, CBFE has thus far only been introduced for discrete active inference. Therefore, in this paper, we used the BFE.

To make message passing-based inference easy for the application engineer, there exist software toolboxes that have pre-computed message update rules

for common factors and common distribution types [1, 16]. In principle, these toolboxes automate the inference process by calling pre-computed update rules.

Unfortunately, the sum-product rule is not always analytically solvable to a closed-form expression. In the next two sub-sections, we will discuss alternative message computation rules (Variational Message Passing and the Unscented Transform) that mesh seamlessly with the sum-product rule, leading to a *hybrid* message passing inference process. The interested reader is referred to [14] for a more in-depth explanation of message passing.

4.3 Variational message passing

The sum-product rule leads to closed-form outgoing messages if all incoming messages are Gaussian and the factor is a linear transformation. However, the graph for the controller's model contains a few factors where these conditions are not met. In those cases, Variational Message Passing (VMP) often resolves the issue since the VMP message computation rules lead to closed-form updates for all distributions in the exponential family as long as conjugacy is maintained [8]. VMP is a message-passing implementation of the more general variational approach to Bayesian inference [8]. Variational inference minimizes an upper bound (the variational free energy) on Bayesian evidence. In this way, the hard problem of evaluating an integral (needed for the Bayes rule) is replaced by an easier optimization problem.

Technically, for any node $f(y, x_1, \ldots, x_n)$, the VMP rule for an outgoing message over edge y is given by

$$\overrightarrow{\mu}(y) = \exp\Big(\int \underbrace{\overrightarrow{\mu}(x_1)..\overrightarrow{\mu}(x_n)}_{\text{incoming messages}} \underbrace{\ln f(y, x_1, .., x_n)}_{\text{log node function}} \mathrm{d}x_1..\mathrm{d}x_n\Big). \tag{13}$$

As an example of hybrid sum-product and VMP message passing-based inference, consider updating Bayesian beliefs about the measurement covariance matrix R_t, given a prior belief $q(R_{t-1})$ and a new observation y_t, see also Figure 1. Let the message

$$\overrightarrow{\mu}(R_{t-1}) = q(R_{t-1}) = \mathcal{W}^{-1}(R_{t-1} \mid \overrightarrow{n}_{t-1}, \overrightarrow{V}_{t-1}) \tag{14}$$

denote the posterior belief about R after observing $y_{1:t-1}$. This message will be used as the prior belief for time step t and is passed to the indicated equality node in Figure 1. The equality node also receives a message from the connected Gaussian node above. This message can be computed by the sum-product rule,

$$\overleftarrow{\mu}(R_t) = \int \overleftarrow{\mu}(v_t)\mathcal{N}(v_t \mid 0, R_t)\mathrm{d}v_t \propto \mathcal{W}^{-1}(R_t \mid \overleftarrow{n}_t, \overleftarrow{V}_t). \tag{15}$$

Note that the computation of $\overleftarrow{\mu}(R_t)$ uses an incoming message $\overleftarrow{\mu}(v_t)$ from the addition node. The equality node processes the two incoming messages to

an updated posterior as follows:

$$q(R_t) = \int \overrightarrow{\mu}(R_{t-1})\overleftarrow{\mu}(R_t)\overbrace{f_=(R_{t-1}, R_t)}^{\text{equality node}}\mathrm{d}R_{t-1} \propto \mathcal{W}^{-1}\big(R_t \mid n_t, V_t\big),$$

and the forward message from the measurement noise can be computed by a VMP update:

$$\overrightarrow{\mu}(v_t) \propto \exp\Big(\mathbb{E}_{q(R_t)}\left[\ln p(v_t \mid R_t)\right]\Big) \propto \mathcal{N}(v_t \mid 0, n_t V_t)\,. \tag{16}$$

4.4 Non-linear Dynamics and the Unscented Transform

In the controller's model, the transition function $g(s_t, u_t)$ is non-linear. When Gaussian messages are passed through a non-linear function, the outgoing message is non-Gaussian, both for the sum-product and VMP update rules. To keep going, we need to project the outgoing message in some way back to a Gaussian distribution.

Here, we discuss using the Unscented Transform (UT) to approximate outgoing messages with normal distributions [18]. As an example, consider the outgoing message $\overrightarrow{\mu}(s_{t+1})$ for the transition node with incoming messages

$$\overrightarrow{\mu}(u_{t+1}) = \mathcal{N}(u_{t+1} \mid \overrightarrow{m}^u_{t+1}, \overrightarrow{P}^u_{t+1})\,, \quad \overrightarrow{\mu}(s_t) = \mathcal{N}(s_t \mid \overrightarrow{m}^s_t, \overrightarrow{P}^s_t)\,, \tag{17}$$

as illustrated in Figure 1.

The Unscented Transform starts by selecting a set of "sigma points" $x_t^{(i)}$ and weights $\omega^{(i)}$ for $i = -M, \ldots, M$. For more details on the computation of sigma points and the weights, we refer to [7]. Next, the sigma points $x_t^{(i)}$ are processed through the nonlinear function as $\xi_t^{(i)} = g(x_t^{(i)})$. Then we compute the parameters of the outgoing (Gaussian) message $\overrightarrow{\mu}(s_{t+1}) = \mathcal{N}(s_{t+1} | \overrightarrow{m}^s_{t+1}, \overrightarrow{P}^s_{t+1})$ with mean and covariance matrix as

$$\overrightarrow{m}^s_{t+1} = \sum_{i=-M}^{M} \omega^{(i)}\xi_t^{(i)}\,,\ \overrightarrow{P}^s_{t+1} = \sum_{i=-M}^{M} \omega^{(i)}(\xi_t^{(i)} - \overrightarrow{m}^s_{t+1})(\xi_t^{(i)} - \overrightarrow{m}^s_{t+1})^\top. \tag{18}$$

4.5 Specification and Inference for the Control Prior

Real-world applications are often subject to environmental constraints, such as limits on engine power. In our application, We are interested in setting a prior constraint on the control signal in the form of $a < u_t < b$, where $a \in \mathbb{R}$ and $b \in \mathbb{R}$. While such a prior looks non-Gaussian, [10] describes an interesting way to implement these kinds of constraints efficiently by Gaussian message passing in a Normal distribution with Unknown Variance (NUV). A box-NUV prior is specified by a probabilistic sub-model, which contains two Normal distributions

with means a and b, and unknown variances $\sigma_a^2 \in \mathbb{R}^+$ and $\sigma_b^2 \in \mathbb{R}^+$ with Gamma distribution priors. The box-NUV prior is specified as

$$p(u_t, \sigma_a^2, \sigma_b^2) = \mathcal{N}(u_t \mid a, \sigma_a^2)\mathcal{N}(u_t \mid b, \sigma_b^2)\Gamma\left(\sigma_a^2 \mid \frac{3}{2}, \frac{\gamma^2}{2}\right)\Gamma\left(\sigma_b^2 \mid \frac{3}{2}, \frac{\gamma^2}{2}\right), \quad (19)$$

where $\Gamma(\cdot|\alpha, \beta)$ represents a Gamma distribution with shape and rate parameters α and β, respectively. As it is shown in [9] the box-NUV prior can be obtained by:

$$\tilde{p}(u) = \sup_{\sigma_a^2, \sigma_b^2} p(u, \sigma_a^2, \sigma_b^2) \propto \exp\left\{-\gamma(|u-a| + |u-b|)\right\}. \quad (20)$$

In this paper for finding the σ_a^2, σ_b^2 we use the expectation maximization update rules according to [2].

5 Experiments

5.1 Experimental Setup

We evaluate the performance of the proposed controller. All experiments were simulated using the Julia programming language on a laptop with an Intel Core i9-12900HK processor and 32 GB of DDR4 RAM. To implement the controller, we used the open-source Julia package RxInfer [1], which supports (variational) Bayesian inference in models through hybrid message passing on an FFG. RxInfer implements many message passing techniques, including the needed methods that we discussed in this paper, namely the sum-product rule for Gaussian messages, VMP for conjugate distributions, the Unscented Transform for non-linear factors, and NUV priors for the control signal. In terms of trustworthiness, RxInfer comes with a large set of unit tests and has previously been used successfully in a wide range of applications, including audio processing devices [17] and control tasks [12]. In all simulations, we assume the Cart-Pole mechanical system described in (2), with parameter values $m_c = 1$ (kg), $m_p = 1$ (kg), $g = 9.81$ (m/s^2), $l = 0.5$ (m), and $\Delta t = 0.01$ (sec).

5.2 Controlling the Cart-Pole System

We validate the performance of the introduced model in section 3 for controlling the Cart-Pole system. For the controller, we used the model as described in (6)-(11). The parameters were set to $T = 100$, $a = -100$, $b = 100$, $Q = 10^{-8}$, $\gamma = 200$, $V_0 = 0.1 \cdot I$, and $n_0 = 10$. We set the controller time step size the same as the Cart-Pole system, i.e., $\Delta t = 0.01$ (sec).

We add noise with a variance $R^* = 0.01 \cdot I$ to the observation at each time step. The system starts in the state $s_0 = [0, -\pi, 0, 0]$ and the goal state is $s_T = [0, 0, 0, 0]$.

Figure 2 illustrates the evolution of the angle θ_t, position x_t, and estimated noise variance R_t during 800 time steps, totaling 8 seconds. It can be seen that the controller successfully guides the system to the target position. For R, we used the mode of $q(R)$, which for both dimensions converges to 0.015, a good approximation of the true added measurement noise.

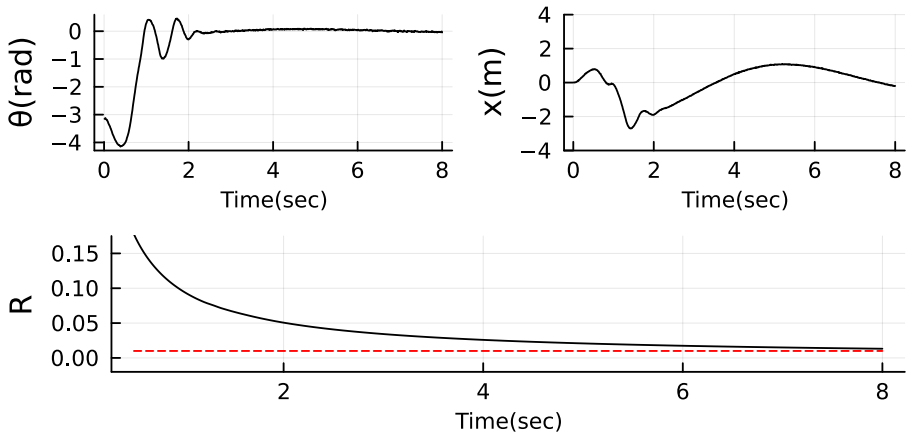


Fig. 2: Evolution of the pole angle (top-left), cart position (top-right), and inferred noise variance (bottom) over time. The dashed red line is the true added noise (R* = 0.01).

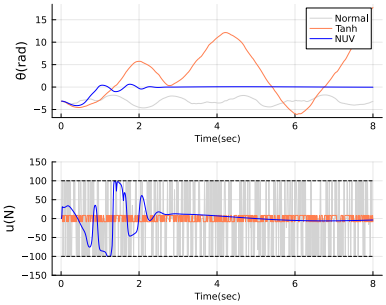
5.3 Alternative Control Limiters

In this experiment, we change the generative model of the controller and measure the effect on its performance. In particular, we compare using the box-NUV node with two alternative ways of limiting the control signal.

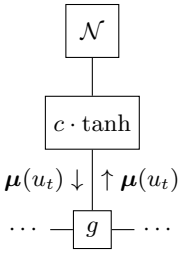
In the first alternative model, we set the action prior $p(u) = \mathcal{N}(0, c)$, where $c = 100$ is the control limit. This is the simplest model assumption. In the second alternative, we set the control prior as $p(u) = \mathcal{N}(0, c)$, and also we use a tanh-node according to the figure 3b. Using the tanh function is motivated by the fact that a scaled tanh function is a commonly used limiter, which, for instance, has been successfully used as an action constraint in [12]. But in our application, using the tanh function leads to a problem in the backward message. Since in the backward direction, the tanh-node acts like $\tanh^{-1}$ and its input should be mapped into $(-1, 1)$. This can be achieved by scaling the input as $(u_t - \delta)/c$, where δ is a small number used to prevent hitting the asymptotes 1 and -1 numerically. But, even with a tiny $\delta = 10^{-14}$ the $\tanh^{-1}$ range leads to around $[-18, 18]$ since $\tanh^{-1}((100 - 10^{-14})/100) = 18.7$. This dramatically reduces the controller's performance. The third approach is using the box-NUV node discussed in 4.5. The performance of the three controllers is shown in Figure 3a. Clearly, only the box-NUV node leads to the desired behavior of the controller.

5.4 Alternative Dynamics Approximations

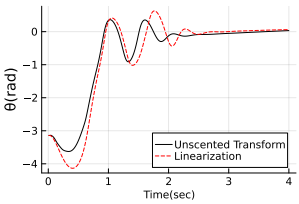
We tested an alternative procedure to UT for passing Gaussian messages through the non-linear transition function $g(s_t, u_t)$. Linearization refers to a first-order Taylor approximation of the nonlinear dynamics around an operating point. We



(a) Comparison of the evolution of pole angle θ_t and inferred control u_t for three different control priors: Normal prior, tanh constraint, and box-NUV prior.



(b) The FFG of setting the limits on controls using the tanh transformation.



(c) Comparison of the evolution of the pole angle for the Unscented Transform and Linearization methods.

applied this method to approximate each of the outgoing messages of the g node. For each, the operating points were the means of the incoming messages. As illustrated in Figure 3c, the agent reaches the goal sooner. We may thus conclude that UT is a more useful approximation.

6 Conclusions

We introduced a fully Bayesian controller for a cart-pole system. While the Bayesian approach to control has a reputation for being both conceptually and computationally challenging, our findings demonstrate a viable path forward. By leveraging the active inference framework and employing a fast message passing-based inference toolbox, we showed how the role of the application engineer predominantly involves specifying a (biased) generative model for the controller. The clear separation of model specification from the inference process offers numerous benefits, notably streamlining the coding process, with the typical generative model requiring no more than half a page of code, even for complex controllers. Moreover, by automating the inference process, the application engineer can divert focus from computational efficiency issues, with this responsibility resting with the designers of the inference toolbox. In light of these advancements, we anticipate a growing prominence of Bayesian control applications.

Acknowledgments. This work was carried out in the context of the BayesBrain project. We gratefully acknowledge financial support from the Eindhoven Artificial Intelligence Systems Institute (EAISI) at TU Eindhoven.

Disclosure of Interests. The authors declare no conflict of interest.

References

1. Bagaev, D., de Vries, B.: Reactive message passing for scalable Bayesian inference. Scientific Programming (2023)
2. van Erp, B., Bagaev, D., Podusenko, A., İsmail, Ş., de Vries Bert: Multi-agent trajectory planning with NUV priors. American Control Conference (2024, in press)
3. Esaki, K., Matsumura, T., Minusa, S., Shao, Y., Yoshimura, C., Mizuno, H.: Dynamical perception-action loop formation with developmental embodiment for hierarchical active inference. In: International Workshop on Active Inference. pp. 14–28. Springer (2023)
4. Friston, K.: What is optimal about motor control? Neuron **72**(3), 488–498 (2011)
5. Friston, K., Samothrakis, S., Montague, R.: Active inference and agency: optimal control without cost functions. Biological Cybernetics **106**, 523–541 (2012)
6. Friston, K.J., Parr, T., de Vries, B.: The graphical brain: belief propagation and active inference. Network neuroscience **1**(4), 381–414 (2017)
7. Gustafsson, F., Hendeby, G.: Some relations between extended and unscented Kalman filters. IEEE Transactions on Signal Processing **60**(2), 545–555 (2011)
8. Jordan, M.I., Ghahramani, Z., Jaakkola, T.S., Saul, L.K.: An introduction to variational methods for graphical models. Machine learning **37**, 183–233 (1999)
9. Keusch, R.: Composite NUV priors and applications. Doctoral Thesis, ETH Zurich (2022). https://doi.org/10.3929/ethz-b-000575227, iSBN: 9783866287686
10. Keusch, R., Loeliger, H.A.: Model-predictive control with new NUV priors. arXiv preprint arXiv:2303.15806 (2023)
11. van de Laar, T., Koudahl, M., van Erp, B., de Vries, B.: Active inference and epistemic value in graphical models. Frontiers in Robotics and AI **9**, 794464 (2022)
12. van de Laar, T.W., de Vries, B.: Simulating active inference processes by message passing. Frontiers in Robotics and AI **6**, 20 (2019)
13. Lanillos, P., Meo, C., Pezzato, C., Meera, A.A., Baioumy, M., Ohata, W., Tschantz, A., Millidge, B., Wisse, M., Buckley, C.L., et al.: Active inference in robotics and artificial agents: Survey and challenges. arXiv preprint arXiv:2112.01871 (2021)
14. Loeliger, H.A., Dauwels, J., Hu, J., Korl, S., Ping, L., Kschischang, F.R.: The factor graph approach to model-based signal processing. Proceedings of the IEEE **95**(6), 1295–1322 (2007)
15. Mills, A., Wills, A., Ninness, B.: Nonlinear model predictive control of an inverted pendulum. In: American control conference. pp. 2335–2340. IEEE (2009)
16. Minka, T., Winn, J., Guiver, J., Zaykov, Y., Fabian, D., Bronskill, J.: /Infer.NET 0.3 (2018), microsoft Research Cambridge. http://dotnet.github.io/infer
17. Podusenko, A., van Erp, B., Koudahl, M., de Vries, B.: AIDA: An active inference-based design agent for audio processing algorithms. Frontiers in Signal Processing **2**, 842477 (2022)
18. Särkkä, S., Svensson, L.: Bayesian filtering and smoothing, vol. 17. Cambridge University Press (2023)
19. Smith, R., Friston, K.J., Whyte, C.J.: A step-by-step tutorial on active inference and its application to empirical data. Journal of mathematical psychology **107**, 102632 (2022)
20. Stengel, R.F.: Optimal control and estimation. Courier Corporation (1994)
21. Tedrake, R.: Underactuated robotics: Learning, planning, and control for efficient and agile machines. Course notes for MIT **6**, 832 (2009)
22. Tipping, M.E.: Sparse Bayesian learning and the relevance vector machine. Journal of Machine Learning Research **1**, 211–244 (2001)

23. Yedidia, J.S., Freeman, W.T., Weiss, Y.: Constructing free-energy approximations and generalized belief propagation algorithms. IEEE Transactions on information theory **51**(7), 2282–2312 (2005)

Belief sharing: a blessing or a curse

Ozan Çatal[1], Toon Van de Maele[1], Riddhi J. Pitliya[1,2], Mahault Albarracin[1,3], Candice Pattisapu, and Tim Verbelen[1]

[1] VERSES Research Lab, Los Angeles, California, 90016, USA
[2] Department of Experimental Psychology, University of Oxford, Oxford, UK
[3] Department of Computer Science, Université du Québec à Montréal, Montréal, Canada
ozan.catal@verses.ai

Abstract. When collaborating with multiple parties, communicating relevant information is of utmost importance to efficiently completing the tasks at hand. Under active inference, communication can be cast as sharing beliefs between free-energy minimizing agents, where one agent's beliefs get transformed into an observation modality for the other. However, the best approach for transforming beliefs into observations remains an open question. In this paper, we demonstrate that naively sharing posterior beliefs can give rise to the negative social dynamics of echo chambers and self-doubt. We propose an alternate belief sharing strategy which mitigates these issues.

Keywords: Active inference · Belief sharing · Multi-agent systems.

1 Introduction

Communication and the emergence of language have been a cornerstone in the development of human intelligence, as they enable human collaboration at multiple communal scales [1]. This collaborative capability hinges significantly on how agents share and process information, particularly requiring their internal beliefs about the world to be aligned [2, 3]. Active inference provides a compelling framework for understanding and designing such collaborative interactions in the interest of building ecosystems of intelligence [4–6]. In this paradigm, agents minimize variational free energy [7], as each agent maintains a generative model of the world which it uses to make inferences about hidden states and to plan actions. Then, communication between agents at the lowest level can be conceptualized as sharing these internal beliefs, transforming the beliefs of one agent into observable data for another [8], these beliefs can be shared directly as we will demonstrate in this paper, but could also be present in the environment more permanently in the form of scripts[9] and texts [10, 11]. This belief-sharing mechanism is intended to facilitate a more coherent and efficient joint exploration of the environment.

However, translating and sharing beliefs has its challenges, as the messages shared are typically colored by one's personal priors and biases [12]. We found

that naively sharing posterior beliefs can inadvertently lead to detrimental social dynamics, such as echo chambers, in which agents reinforce each other's biases, and self-doubt, in which agents discount their observations to favor shared, yet incorrect, beliefs. These phenomena can significantly impair the collective performance of the agents, highlighting the need for more sophisticated strategies in belief communication. In this paper, we explore these dynamics in depth. We begin by modeling the communication between agents as belief sharing under the active inference framework, demonstrating the pitfalls of straightforward belief sharing. We then propose an alternative strategy that mitigates these issues by adjusting how beliefs are communicated. Specifically, we advocate for sharing likelihood information rather than posterior beliefs, treating other agents' observations as additional independent sources of information. This approach aims to harness the benefits of collaborative inference while avoiding the pitfalls of misleading belief reinforcement.

Our contributions are threefold: (i) We provide a detailed analysis of how naive belief-sharing can lead to echo chambers and self-doubt. (ii) We propose a novel communication strategy that mitigates these issues by sharing likelihoods. (iii) We validate our approach through simulations, demonstrating improved performance and robustness in collaborative tasks. The following sections outline our active inference model for communication, describe the experimental setup used to test our hypotheses, present our findings on echo chambers and self-doubt, and discuss our proposed solution and its implications for designing collaborative AI systems.

2 An active inference model for communication

In this section, we provide a summary overview of active inference, and how it can be adopted to model communication between agents. For a more in depth overview of active inference we refer the reader to [7].

2.1 Perception and planning as inference

Active inference posits that agents entertain a generative model of the environment they operate in, and casts perception and action as Bayesian inference [7]. In general, the agent's generative model can be written as the joint probability distribution over states s, observations o and actions a, with tilde denoting a time sequence of those over timesteps t:

$$P(\tilde{s}, \tilde{o}, \tilde{a}) = P(s_0) \prod_t P(o_t|s_t) P(s_t|s_{t-1}, a_{t-1}) P(a_{t-1}) \tag{1}$$

Perception now becomes inferring the posterior distributions of states given the performed actions and observations. As this is typically intractable, agents resort to variational Bayesian inference, where an approximate posterior $Q(\tilde{s}|\tilde{o})$ is optimized instead, by minimizing the variational Free Energy:

$$\begin{aligned} F &= \underbrace{D_{KL}[Q(\tilde{s}|\tilde{o}))||P(\tilde{s}|\tilde{a},\tilde{o})]}_{\text{posterior approximation}} - \underbrace{\log P(\tilde{o})}_{\text{log evidence}} \\ &= \underbrace{D_{KL}[Q(\tilde{s}|\tilde{o}))||P(\tilde{s},\tilde{a})]}_{\text{complexity}} - \underbrace{\mathbb{E}_{Q(\tilde{s}|\tilde{o})}[\log P(\tilde{o}|\tilde{s})]}_{\text{accuracy}} \end{aligned} \tag{2}$$

It is clear that minimizing variational Free Energy is equivalent with maximizing a bound on the (log) evidence or ELBO [13], and encourages the model to maximize accuracy with minimal complexity.

To interact with the environment, an agent also needs to select a sequence of actions or policy $\pi = \{a_t, a_{t+1}, ...\}$ to execute. In active inference, planning is also treated as inference, assuming that agents will prefer policies that minimize expected Free Energy G. More specifically, policies are selected from

$$\begin{aligned} &P(\pi) = \sigma(-G(\pi)), \text{ with} \\ &G(\pi) = \sum_{\tau=t+1}^{T} \underbrace{\mathbb{E}_{Q(o_\tau|\pi)}\big[D_{KL}[Q(s_\tau|o_\tau,\pi)||Q(s_\tau|\pi)]\big]}_{\text{(negative) Information Gain}} - \underbrace{\mathbb{E}_{Q(o_\tau|\pi)}\big[\log P(o_\tau)\big]}_{\text{Utility}} \end{aligned} \tag{3}$$

Here, σ denotes the softmax function, and the expected Free Energy balances information gain with some prior preference distribution over future outcomes or utility.

2.2 Communication as belief sharing

When agents share a common world and world model, they can benefit from sharing beliefs among each other [8]. The most straightforward way to realize this would be to share the agent's respective posterior beliefs on some shared modality. In order to achieve this, the agents generative model is expanded as shown in Fig. 1. In particular, any agent, for example., the primary focal agent, assumes other agents with a similar generative model will communicate information about its beliefs of the world. To do so, we equip the focal agent with an extra observation modality o_t^s. Instead of being observed from the environment, o_t^s is an observation generated by another agent based on its internal beliefs s'_t. This approach is the kind of model posited in earlier work [8].

To realize posterior belief-sharing, agents require a likelihood mapping between posterior beliefs about latent states that are shareable among agents, i.e., s_t, and this observation modality o_t^s. In natural systems, these can be a very complex likelihood mapping, e.g., language [14] or birdsong [15]. However, in the case of AI agents, we can decide on the communication channel ourselves. One particular naive choice is to directly share the sufficient statistics of one's internal beliefs s'_t and integrate these with an identity likelihood mapping, as used in [8]. However, in the remainder of this paper, we will demonstrate the fallacies of using this approach and propose a different format for shared messages.

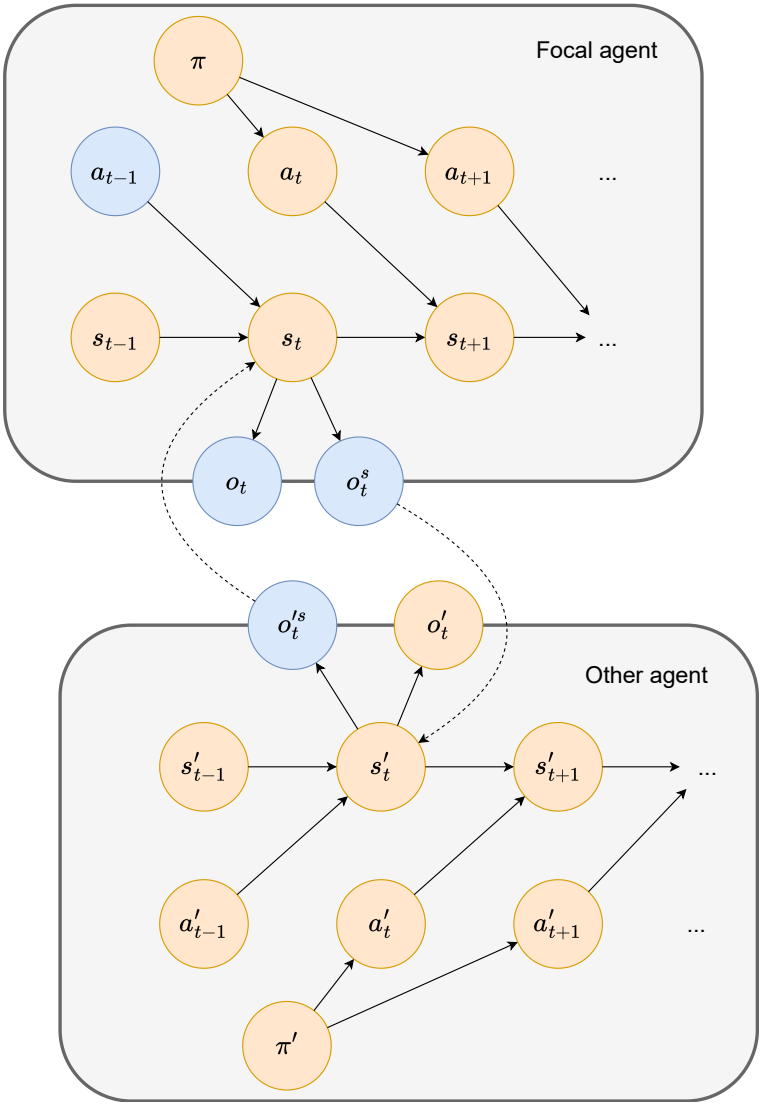
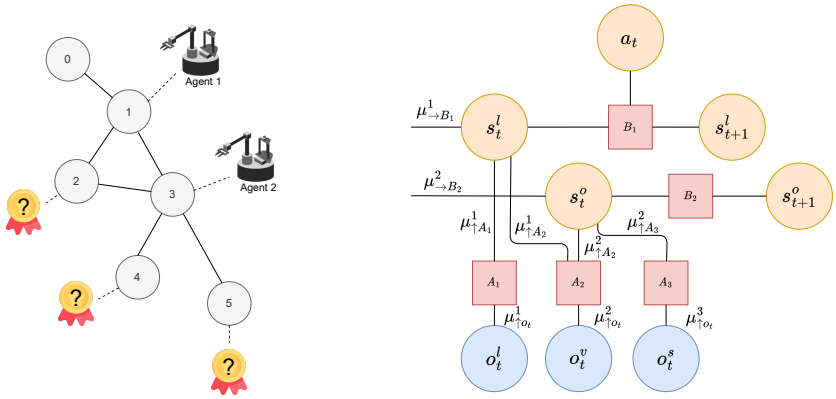


Fig. 1: **Two active inference agents sharing beliefs.** The generative model of each agent is a POMDP, where observations are generated from a hidden state s_t. Actions a_t, generated from a policy π, transition this state. In addition to the typical observation o_t at each timestep t, each agent also receives a shared observation o_t^s that is generated from the other agent's internal beliefs s'_t. Blue variables are observed from the perspective of the focal agent, i.e. they observe their own actions, observations and the observations shared with the other agent.



(a) (b)

Fig. 2: **Illustration of the graph environment and the agent's factor graph**. (a) agents are located on a connected graph of locations and need to find a rewarding object that might be present at one of the locations. (b) a factor graph representation of the agent's generative model. Two latent state factors that model the agent's location and the object's location respectively, give rise to two sensory modalities through a likelihood factor: the agent's location (A_1) and whether the object is visible (A_2). In addition, agents can share beliefs about the object location through belief sharing (A_3). The agent's location can change conditioned on move actions (B_1), whereas the object is kept static in our experiments ($B_2 = I$).

3 Experimental setup

To demonstrate multi-agent belief sharing, we simulate an object-finding task, where multiple agents search for a rewarding object in the same environment, and can potentially share beliefs on where they think the object is. The setup and generative model for this task is depicted in Figure 2. The world is represented by a graph of N locations that can be visited by the agents, and agents can move between connected nodes in the graph. Each agent has two state factors, a Categorical(N) variable s^l_t which is the belief about the agent's location, and a Categorical(N) variable s^o_t which is the belief about the object location. An agent can perform one of N move actions a, modeled using the three-dimensional dynamics tensor B_1.

$$B_1^{i,j,a} = \begin{cases} 1.0, & \text{if } a = i \wedge \text{connected}(i,j) \\ 1.0, & \text{if } i = j \\ 0.0, & \text{otherwise} \end{cases}$$

Where the indices i and j indicate the i-th and j-th location in the environment. We assume the object is static, i.e. its dynamics B_2 are modeled by the identity matrix I. The agent has three observation modalities. First, it observes its location o_t^l, with a near identity likelihood mapping to the location state factor:

$$A_1^{i,j} = \begin{cases} 0.99, & \text{if } i = j \\ 0.01, & \text{otherwise} \end{cases}$$

Second, it observes o_t^v whether the object is visible or not. This is governed by a three-dimensional likelihood mapping A_2 where

$$A_2^{v,i,j} = \begin{cases} 0.2, & \text{if } i = j \wedge v = \text{not visible} \\ 0.8, & \text{if } i \neq j \wedge v = \text{not visible} \\ 0.8, & \text{if } i = j \wedge v = \text{visible} \\ 0.2, & \text{if } i \neq j \wedge v = \text{visible} \end{cases}$$

Finally, there is the belief-sharing observation o_t^s which contains the shared information from the other agent.

The agents are initialized with a prior on s_t^l set to their starting location, and a prior on s_t^o set to the initial belief on where to find the object. This is typically set uniform to (a subset of) the available locations to foster searching behavior. The preference C is to have the object visible outcome.

4 Echo chambers

In a model which shares the posterior beliefs of one agent as observations of another, Bayesian model updating reinforces redundant priors shared among them. The consequent simulated behavior mirrors the “echo chamber effect" wherein messages communicated by like-minded agents are amplified and returned. Psychological interpretations of the echo chamber effect are illustrated in the consequences of social media feed algorithms, which are often engineered to encourage user engagement with sympathetic posts [16]. An algorithmically curated social media curriculum increases engagement by anticipating a user's expected social media observations and fulfilling those expectations. Promoted content thereby constructs homophilic interaction networks which facilitate the construction and reinforcement of shared narratives among users. In worst case scenarios, the result is the unimpeded flow of misinformation on social media platforms. More generally, shared narratives facilitated by social media feed algorithms result in an increase in confidence of posterior beliefs even when external evidence is absent or intentionally excluded.

Ignoring new evidence can be adaptive, such as when this strategy facilitates in-group cooperation [17]. However, negative consequences also result, such as when outside sources are discredited or distrusted.

Corresponding to the echo chamber effect, one pitfall of belief sharing is that when agents have established even small prior beliefs on their goal, if those priors

are shared, then an echo chamber forms. The agents, however, reinforce their prior beliefs through the communication method, resulting in ever-increasing beliefs on the goal location even when there is no new evidence to support this belief. Fig. 3 shows a simulation triggering this situation. The agents start with a small belief that the object will be present at two locations within the graph to simulate a longer-running experiment where such a situation might naturally occur. We have restricted the movement of the agents and prohibited the agents from accumulating more evidence about the object's actual location. However, the agents keep increasing their belief on the object being present at the a priori believed location because of the constant sharing of beliefs. Once the agents create such an echo chamber, more often than not, they are stuck in this faulty belief unless they all sufficiently change their belief at the same time.

5 Self-doubt

In an echo chamber, a belief sharing agent's confidence about their priors is reinforced in the absence of external evidence. Conversely, self-doubt refers to a scenario in which the agent's self-confidence is degraded as a result of belief sharing. In our simulations, the paradigmatic example is when multiple agents are fixated on a search task with complimentary plans for exploring the envi-

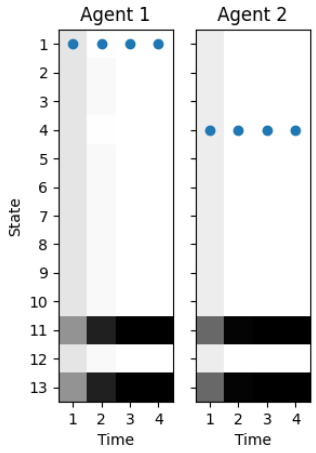


Fig. 3: **Simulation of an echo-chamber**. We initialize both agents with a small prior belief that the object will be present at location 11 or 13. Then, we let the agents share their beliefs. Note that this reinforces the belief that the object will be at either one of the locations. The next columns show the evolution of both agents where they keep observing the environment. We see that in the transition from time 1 to time 2 the agents increase the belief that the object is at the a priori believed location even though there is no new evidence to support this belief. Agent location is depicted using the blue dots in both panels.

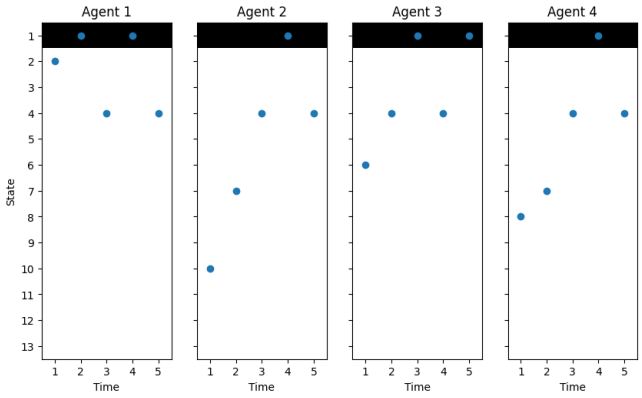


Fig. 4: **Simulation of self-doubt for 4 agents.** Again, each panel displays the evolution of the posterior belief as a function of time, where darker colors indicates a higher degree of belief. Blue dots indicate the agent location. All agents are initialized with a strong prior belief that the object will be at location 1, as indicated by the dark shaded area. This reflects a potential situation where all agents have been acting in the environment for a long time, accruing faulty evidence. In this case the communication mechanism prohibits the agents from discovering that the object is not there, even after observing its absence multiple times.

ronment. If multiple agents participate in the search task, they can get stuck on faulty beliefs and ignore all evidence that points to the contrary. This can occur after an echo chamber is formed but could also occur independently. In this scenario, the agents again reinforce each other's beliefs in such a strong way that the agents "doubt" their observations originating from the environment. Fig. 4 visually overviews a simulation showcasing this phenomenon. The agents are initialized with a strong belief on the object location. In this particular simulation, the agents have a single peaked belief on the object location (location 1 in the graph), which could occur after an echo chamber situation. The agents are unrestricted in their movement as long as they follow the underlying graph structure. We see that even though eventually all agents visit location 1, they cannot correctly eliminate that location as a possible object location. The incoming beliefs of the other agents overrule their sensory observations.

6 To share or not to share?

Given that sharing agents' beliefs can give rise to the aforementioned issues, it is worth wondering what information can best be shared to allow multi-agent cooperation in active inference agents. In particular, the update rule for s_t^o,

written in variational message passing notation [18], is of the form

$$\begin{aligned} s_t^o &= \sigma(\mu^2_{\rightarrow B_2} + \mu^2_{\uparrow A_2} + \mu^2_{\uparrow A_3}), \\ \mu^f_{\uparrow A_g} &= o_t^g \odot \varphi(\mathbf{a}^g) \odot_{i \in pa(g) \setminus f} s_t^i, \end{aligned} \tag{4}$$

where σ is the softmax function, $\mu^f_{\uparrow A_g}$ the message from observation modality g to state factor f and $\varphi(\mathbf{a}^g)$ the digamma function of the Dirichlet counts corresponding to the parameters of the likelihood model [8]. Effectively, the update message is comprised of a part coming from a prior given by our beliefs on the previous timestep $\mu^2_{\rightarrow B_2}$, a part based on the latest observation $\mu^2_{\uparrow A_2}$, and a part communicated by the other $\mu^2_{\uparrow A_3}$. When we communicate the other's posterior parameters directly through an identity likelihood mapping, we have $\mu^2_{\uparrow A_3} = \mu^{2,other}_{\rightarrow B_2} + \mu^{2,other}_{\uparrow A_2}$. This shows that, indeed, when agents have similar priors, this gets double counted in the belief update.

To address this, we will now instead share the other's likelihood message only, i.e. $\mu^2_{\uparrow A_3} = \mu^{2,other}_{\uparrow A_2}$. This scheme leaves out agents' prior beliefs about the state and only shares the agent's interpretation of the observation, treating the other agents as extra independent observers for the exact latent cause in the world. We call this scheme 'likelihood sharing'.

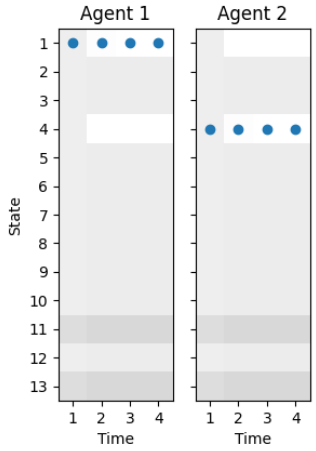


Fig. 5: **Illustration of the lack of echo-chamber like behaviour when sharing likelihoods.** In this figure, the situation leading to an echo chamber is recreated. Both agents are initialized with the same prior belief that the object is most likely at locations 11 and 13; however, because of the sharing of likelihoods, they do not get stuck in an echo chamber and do not increase the beliefs when there is no new evidence.

In the particular scenario of object-finding agents in a graph world, the agents share their current location and visibility observation as passed through their

object location A-tensor; each receiving agent can integrate this observation quite easily into their own posterior belief by using Bayes rule. In effect, each agent treats the other agents as an extra "pair of eyes" in the search for the object, inferring their posterior update if they observed what the other agent had observed.

In Fig. 5 and Fig 6, the same simulations from earlier are reprised but using the new likelihood-sharing mechanism. The echo chamber and self-doubt phenomena are no longer present even when providing the same initial conditions.

Finally, Figure 7 compares likelihood sharing to belief sharing and not communicating at all. All methods are tested over all possible combinations of agent starting locations and object locations in the environment, and each configuration is repeated five times. The trials are evaluated on the percentage of times the object was found after a maximum of 10 timesteps in the environment. From the experiments, the likelihood-sharing agents are on par with the naive belief-sharing agents when they both start from the same uniform prior belief on the object location. However, when both agents start from a non-uniform prior, i.e., a prior where there is 80% belief that the object is at two randomized locations (differing from the agent's initial location and the actual object location), the likelihood sharing agent is better at recovering from these faulty beliefs and finding the object.

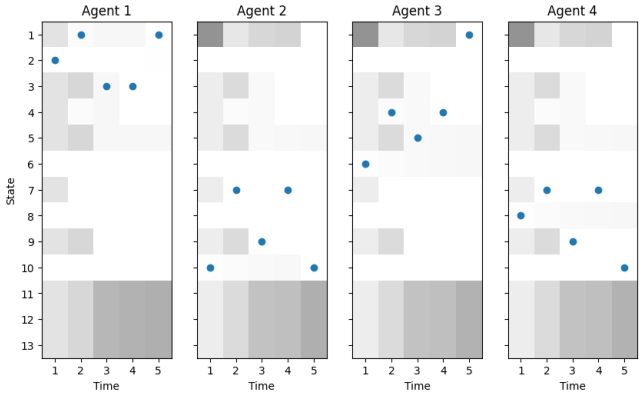


Fig. 6: **Alleviation of the self-doubt behavior under likelihood-sharing.** As with the previous depiction of this scenario, the agents are initialized with a strong belief that the object will be at location 1; nonetheless, due to the different communicated belief, the agent no longer ignores the evidence for the object's absence.

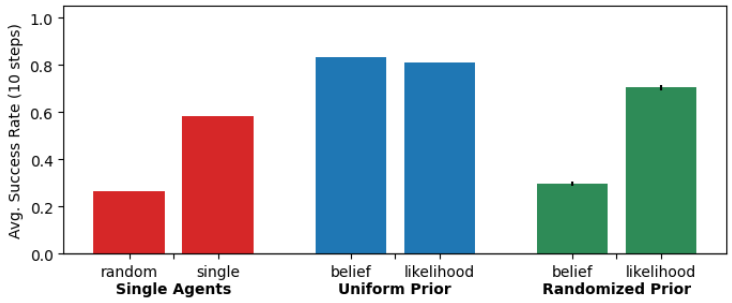


Fig. 7: **Overview of the average object find rate for each type of agent.** We measured the percentage of finding the object after a maximum of 10 timesteps by any of the two agents over all possible start configurations in the environment. The experiments are repeated five times to account for variability in the action selection process and the randomness in initial prior configurations. We report the results for three cases. Single Agents, when the agents do not communicate at all; Uniform Prior, when the agents communicate but each start with a uniform prior belief on the object location; and finally, Randomized Prior, when the agents communicate, but the prior is randomized to prefer two random locations other than the agents own location and the actual location.

7 Discussion

The results presented in this paper highlight the potential pitfalls of belief-sharing in multi-agent systems under the framework of active inference. Our findings suggest that naive sharing of posterior beliefs can lead to undesirable social dynamics such as echo chambers and self-doubt, severely hampering the agents' performance in collaborative tasks.

Echo chambers in human societies are well-documented phenomena where groups of individuals reinforce their preconceptions, often without external validating evidence. In our simulations, a similar effect occurs when agents continuously share their posterior beliefs. Initial biases can get amplified through repetitive belief sharing, leading to overly confident but potentially erroneous shared beliefs. This results in the agents becoming overconfident in incorrect hypotheses, thereby hampering their search or exploration processes. Similarly, self-doubt arises when agents' observations contradict the reinforced shared beliefs, leading them to disregard their sensory inputs. This mirrors real-world psychological effects where individuals question their perceptions in the face of strong peer influence. In our simulations, agents maintained strong incorrect beliefs about the object's location despite direct evidence to the contrary due to the influence of shared but incorrect posterior beliefs.

Our proposed strategy of sharing likelihoods rather than posterior beliefs mitigates the issues of echo chambers and self-doubt. By sharing interpreted observations rather than fully formed beliefs, agents can integrate new information

without being overwhelmed by the potentially erroneous priors of others. This approach allows agents to utilize each other as additional sensing mechanisms, providing independent evidence that can be more robustly combined with their observations. By treating other agents' observations as additional data points rather than beliefs, the system remains more flexible and resilient to individual errors. The proposed approach was only validated in simulated experiments but might also apply to more general and complex scenarios with further modifications.

Our work suggests that the type of information shared among active inference agents must be carefully considered to avoid counterproductive dynamics. Future research can build on these insights by exploring other belief-sharing strategies and their impacts on system performance. Additionally, exploring these dynamics in more complex and varied environments, including those with adversarial elements, could provide deeper insights into the performance of different communication strategies. As our approach assumed full and honest collaboration between the agents, another future avenue of research would be to investigate the impact of dishonesty and, consequently, the discounting of communications between untrusted agents.

In conclusion, while belief sharing among active inference agents can enhance collaborative performance, it also risks reinforcing incorrect beliefs and undermining individual observations. Our proposed likelihood-sharing mechanism offers a promising solution by leveraging the strengths of collective sensing while mitigating the pitfalls of echo chambers and self-doubt without significant changes to the underlying model. Such strategies will be essential for developing robust, efficient, and adaptive collaborative agents as multi-agent active inference systems are designed.

References

1. J. Henrich, *The Secret of Our Success: How Culture Is Driving Human Evolution, Domesticating Our Species, and Making Us Smarter*. Princeton University Press, Oct. 2015.
2. B. Bahrami, K. Olsen, P. E. Latham, A. Roepstorff, G. Rees, and C. D. Frith, "Optimally interacting minds," *Science*, vol. 329, p. 1081–1085, Aug. 2010.
3. A. Constant, M. J. D. Ramstead, S. P. L. Veissière, and K. Friston, "Regimes of expectations: An active inference model of social conformity and human decision making," *Frontiers in Psychology*, vol. 10, Mar. 2019.
4. K. J. Friston, M. J. Ramstead, A. B. Kiefer, A. Tschantz, C. L. Buckley, M. Albarracin, R. J. Pitliya, C. Heins, B. Klein, B. Millidge, D. A. Sakthivadivel, T. St Clere Smithe, M. Koudahl, S. E. Tremblay, C. Petersen, K. Fung, J. G. Fox, S. Swanson, D. Mapes, and G. René, "Designing ecosystems of intelligence from first principles," *Collective Intelligence*, vol. 3, Jan. 2024.
5. K. J. Firston and C. D. Firth, "Active inference, communications and hermeneutics," *Cortex*, vol. 68, pp. 129–143, 2015.
6. R. Tison and P. Poirier, "Active inference and cooperative communication: An ecological alternative to the alignment view," *Frontiers in Psychology*, vol. 12, 2021.

7. T. Parr, G. Pezzulo, and K. J. Friston, *Active Inference: The Free Energy Principle in Mind, Brain, and Behavior*. The MIT Press, 03 2022.
8. K. J. Friston, T. Parr, C. Heins, A. Constant, D. Friedman, T. Isomura, C. Fields, T. Verbelen, M. Ramstead, J. Clippinger, and C. D. Frith, "Federated inference and belief sharing," *Neurosci. Biobehav. Rev.*, p. 105500, Dec. 2023.
9. M. Albarracin, A. Constant, K. J. Friston, and M. J. D. Ramstead, "A variational approach to scripts," *Frontiers in Psychology*, 2021.
10. S. Gallagher and M. Allen, "Active inference, enactivism and the hermeneutics of social cognition," *Synthese*, vol. 195, no. 6, pp. 2627–2648, 2018.
11. N. Bouizegarene, M. Ramstead, A. Constant, K. Friston, and L. Kirmayer, "Narrative as active inference," *Frontiers in Psychology*, vol. 15, 2024.
12. M. Albarracin, D. Demekas, M. J. D. Ramstead, and C. Heins, "Epistemic communities under active inference," *Entropy*, vol. 24, p. 476, mar 2022.
13. C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 1 ed., 2007.
14. 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, p. 42–64, Nov. 2020.
15. K. J. Friston and C. D. Frith, "Active inference, communication and hermeneutics," *Cortex*, vol. 68, p. 129–143, July 2015.
16. M. Cinelli, G. De Francisci Morales, A. Galeazzi, W. Quattrociocchi, and M. Starnini, "The echo chamber effect on social media," *Proceedings of the National Academy of Sciences*, vol. 118, Feb. 2021.
17. M. Kim, B. Park, and L. Young, "The psychology of motivated versus rational impression updating," *Trends in Cognitive Sciences*, vol. 24, p. 101–111, Feb. 2020.
18. J. Winn and C. M. Bishop, "Variational message passing," *Journal of Machine Learning Research*, vol. 6, no. 23, pp. 661–694, 2005.

Exploring and Learning Structure: Active Inference Approach in Navigational Agents

Daria de Tinguy[1[0000−0003−1112−049X]], Tim Verbelen[2], and Bart Dhoedt[1]

[1] Ghent University, Ghent, Belgium first_name.family_name@ugent.be
[2] Verses AI tim.verbelen@verses.ai

Abstract. Drawing inspiration from animal navigation strategies, we introduce a novel computational model for navigation and mapping, rooted in biologically inspired principles. Animals exhibit remarkable navigation abilities by efficiently using memory, imagination, and strategic decision-making to navigate complex and aliased environments. Building on these insights, we integrate traditional cognitive mapping approaches with an Active Inference Framework (AIF) to learn an environment structure in a few steps. Through the incorporation of topological mapping for long-term memory and AIF for navigation planning and structure learning, our model can dynamically apprehend environmental structures and expand its internal map with predicted beliefs during exploration. Comparative experiments with the Clone-Structured Graph (CSCG) model highlight our model's ability to rapidly learn environmental structures in a single episode, with minimal navigation overlap. this is achieved without prior knowledge of the dimensions of the environment or the type of observations, showcasing its robustness and effectiveness in navigating ambiguous environments.

Keywords: exploration · active inference · topological graph · structure learning.

1 Introduction

A functional navigation system must seamlessly fulfil three key functions: self-localisation, mapping, and path planning. This requires both a sensing component for spatial perception and a storage capability to extend these perceptions temporally and spatially [33]. Animals exhibit a remarkable capacity for rapidly learning the structure of their environment, often in just one or a few visits, relying on memory, imagination, and strategic decision-making [32,26].

The hippocampus and neocortex play crucial roles in episodic memory, spatial representation, and relational inference. Mammals rely on mental representations of spatial structures, traditionally viewed as either cognitive maps or cognitive graphs, conceptualising environmental space as a network of nodes [33,25,7,1]. Recent research suggests an integrated approach combining these concepts is more effective [23].

Our approach adopts this viewpoint, proposing a topological map incorporating internal motion (Euclidean parameters) to delineate spatial experiences. The neural positioning system, found in rodents and primates, supports self-localisation and provides a metric for distance and direction between locations [33]. This system includes place cells, heading direction cells [15], grid cells [2], speed cells [14], and border cells [30], working together to enable rapid learning, disambiguation of aliases, and a comprehensive understanding of spatial navigation [5].

Building on these concepts, we introduce a novel model that dynamically learns environmental structure and expands its cognitive map. Integrating visual information and proprioception (inferred body motion), our model constructs locations and connections within its cognitive map. Starting with uncertainty, the model envisions action outcomes, expanding its map by incorporating hypotheses into its generative model, analogous to Bayesian model reduction [9] that grows its model upon receiving new observation, we extend ours upon predicted beliefs. This process allows our model to efficiently navigate and comprehend environmental structures with minimal steps, using an active inference navigation scheme [17]. Compared to the Clone-Structured Graph (CSCG) [24], our model rapidly learns environmental layouts more efficiently. An exploration run is shown in Figure 1, displaying from left to right the full environment, the extracted observation and exploration path and the resulting internal map of the agent.

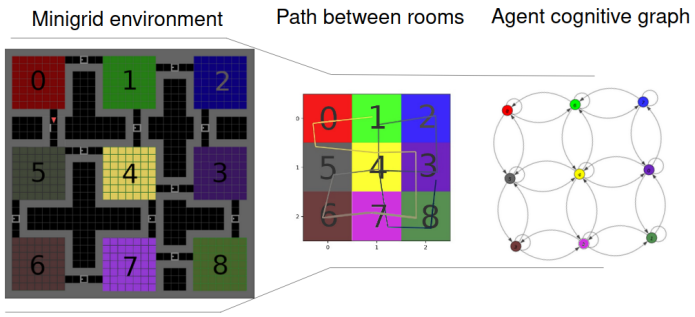


Fig. 1. From the mini-grid environment [3,31] with different rooms annotated by colour to the path our agent took -from black to white- to form a successful exploration correctly linking all the rooms up to the agent's internal topological graph with the state associated to each room.

2 Related work

While navigating their environment, animals often encounter ambiguous sensory inputs due to aliasing, resulting in repetitive observations, such as encountering

two overly similar corridors. They must rapidly disambiguate the structure of their environment to navigate successfully.

Models like the Clone-Structured Graph (CSCG) [12] or Transformers representations [4] have been proposed to form cognitive maps that disambiguate aliased environments through partial observations. However, these models require substantial training time using random or hard-coded policies. In contrast, animals adapt their actions based on subtle cues and incentives, learning to navigate with minimal instances [32].

Animals exhibit decision-making abilities powered by imagination (estimating actions' consequences) and a holistic understanding of the environment, naturally imagining un-visited areas and guiding their next steps [2]. This intuitive decision-making process, considering incentives like food or safety, rapidly directs them toward their objectives [26].

Integrating observations with proprioception [33] helps animals circumvent aliasing, using a process similar to active inference for judgement [17]. Active inference involves continuously updating internal models based on sensory inputs, enabling adaptive and efficient decision-making. This normative framework explains cognitive processing and brain dynamics by positing that actions and perceptions aim to minimise free energy, encapsulating causal relationships among observable outcomes, actions, and hidden states [22,8].

At the core of adaptive behaviour is the balance between exploitation (selecting the most valuable option based on existing beliefs) and exploration (choosing options that facilitate learning) [27]. Recent behavioural evidence suggests humans mix random and goal-directed exploration [11]. Our model adopts this balance through free energy minimisation, choosing stochastic policies to enhance environmental understanding. This enables active learning, rapidly reducing uncertainty over model parameters (i.e. reducing uncertainty over our beliefs) [27].

By balancing curiosity and goal-directed behaviour through free energy, our system guides an agent meaningfully and learns in a biologically plausible way [22]. It achieves few-shot or one-shot learning, similar to mice in a labyrinth [26]. By projecting the consequences of actions into its internal map, the agent extends its imagination beyond known territories, improving its navigation ability to explore environments of any dimension.

3 Method

In our study, our agent initiates exploration of the environment without any prior knowledge regarding the observations and dimensions of the map it is about to navigate. Subsequently, we will clarify how, at each step, the agent engages in inferring the current state, a process that integrates both the notion of observation and proprioception (position perception given a motion). This inference task involves updating past beliefs based on the latest observation and motion, following the principles of a Partially Observable Markov Model (POMDP). Henceforth, the agent strategically envisions sequences of actions to explore, termed policies, while concurrently expanding its internal map to accommodate

potential unexplored areas with uncertain priors. Although the agent may know the relative positions of these areas, it does not foresee observations. This iterative and multi-step process serves as the cornerstone for the agent's adaptive learning and navigation strategies within the environment.

3.1 Inference and spatial abstraction

In the context of Active Inference (AIF), the process of inferring the agent's current state involves integrating sensory inputs and prior beliefs within a Partially Observable Markov Decision Process (POMDP). We consider that our inference mechanism operates at the highest level of abstraction within a hierarchical spatial framework [31], where lower layers handle observation transformation and the concept of blocked paths, akin to how visual observations are processed in the visual cortex and motion limitations are perceived by border cells [30]. Figure 1 illustrates the agent navigating through an environment, where doors signify transitions to different states while walls correspond to obstacles. We give our agent the notion that doors lead to another location, while walls lead to the same observation and the pose stays static. At the centre of this Figure, we see a path taken by the agent depicted along with the observations perceived by our model, demonstrating how observations are simplified and generalised at the highest abstraction level into a single colour per room (floor colour). The internal topological map generated by the agent based on its exploration path is presented in the final frame of Figure 1. The underlying POMDP model guiding this inference process is depicted in Figure 2, where the current state s_t (defining a room) and position p_t (the location of that room) are inferred based on the previous state s_{t-1}, p_{t-1} and action a_{t-1} leading to the current observation o_t (the colour of that room). The generative model capturing this process is described by Equation 1, where the joint probability distribution over time sequences of states, observations, and actions is formulated. Tildes are used to denote sequences over time.

$$P(\tilde{o}, \tilde{s}, \tilde{p}, \tilde{a}) = P(o_0|s_0)P(s_0)P(p_0)P(a_0)\prod_{t=1}^{\tau} P(o_t|s_t)P(s_t, p_t|s_{t-1}, p_{t-1}, a_{t-1}) \tag{1}$$

Due to the posterior distribution over a state becoming intractable in large state spaces, we use variational inference instead. This approach introduces an approximate posterior denoted as $Q(\tilde{s}, \tilde{p}|\tilde{o}, \tilde{a})$ and is presented in equation 2 [28].

$$Q(\tilde{s}, \tilde{p}|\tilde{o}, \tilde{a}) = Q(s_0, p_0|o_0)\prod_{t=1}^{\tau} Q(s_t, p_t|s_{t-1}, p_{t-1}, a_{t-1}, o_t) \tag{2}$$

The classical inference scheme heavily relies on past and current experiences to localise the agent within its environment, using observation alone, the agent would be weak to aliased observations at different locations. By combining observation with the agent's proprioception the model is much more robust in differentiating ambiguous environments. The internal positioning p_0 is initialised

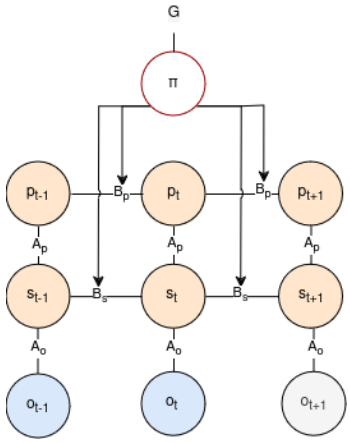


Fig. 2. factor graph POMDP of our generative model transitioning from past and present (up to time-step t) to future (time-step $t+1$). The pose p_t is inferred from the previous pose p_{t-1} and the action from policy π, while the state s_t determined by the corresponding observation o_t and influenced by the previous state s_{t-1}, pose p_t and action a_{t-1}. Past actions and observations are assumed observable, indicated by a blue colour. In the future, the actions are defined by a policy π influencing the new states and position in orange and new predictions in grey.

at the start of exploration in the absence of prior information, and is updated as the agent transitions between rooms (e.i., by passing through a door), thus as long as the agent is confident in its current state. the POMDP factor graph showing the association between poses, states and observations is illustrated in Figure 2

If the agent were to be kidnapped and re-localised elsewhere, the observation o and inferred position p would not match expectations and the confidence in the state would decrease. If the confidence in the state goes below a given threshold, the agent stops updating its internal model given new information and focuses on re-gaining confidence over its state/location.

However, inferring the position p has much more to offer than localisation robustness, it is key to extending the internal map over unexplored areas yet to be integrated into the model through parameter learning.

3.2 Parameter Learning

Learning within the Active Inference framework encompasses the adaptation of beliefs concerning model parameters, such as transition probabilities $P(s_t|s_{t-1})$ (e.g. how rooms are connected) and likelihood probabilities $P(o_t|s_t)$ (e.g. what a room looks like). These parameters reflect the structural connectivity of the environment and the expected sensory outcomes given particular states.

Generative models in Active Inference rely on prior beliefs regarding parameter distributions, with updates driven by the active inference framework [22].

Unlike traditional discrete-time POMDP, where either transitions or likelihoods probabilities are fixed and updating parameters implies reasoning over a fixed spatial dimension [21,13,19], our model learns the probabilities of all its Markov matrices and extends their dimensions dynamically. The state transitions $B_s = P(s_t|s_{t-1}, a_{t-1})$ and the observation $A_o = P(o_t|s_t)$, position likelihood$A_p = P(p_t|s_t)$ probabilities are optimised over transitions. The position transition $B_p = P(p_t|p_{t-1}, a_{t-1})$, however, is not a Markov matrix and entails an incremental process based on consecutive motions (experimented or predicted), without any parameters to be learned by belief optimisation.

The optimisation of beliefs of the generative model parameters θ occur after state inference and involves minimising the free energy F_θ while considering prior beliefs and uncertainties associated with both parameters and policies, as defined in [22]:

$$\begin{aligned} \theta =& (A_o, A_p, B_s) \\ F_\theta =& \mathbb{E}_{Q(\pi,\theta)}[F(\pi, \theta)] + D_{KL}[Q(\theta)||P(\theta)] + D_{KL}[Q(\pi)||P(\pi)] \end{aligned} \tag{3}$$

With P and Q being respectively the joint distribution and the approximate posterior of the model. The model updates its parameters based on observed data and transitions, expanding the observation dimension of A upon encountering new information as can be seen in [29]. At initialisation, high certainty is assigned to the likelihood probabilities integrating the first observation. After realising the parameter update based on priors, the model edits its internal map dimensions and parameters based on predicted transitions, expanding all parameters in their state dimensions, thus improving exploration in unexplored environments of any unknown size.

3.3 Incorporating spatial dynamics in model parameters

To extend the internal map (our state space), we propose a novel approach where the agent predicts one-step policy outcomes in all directions considering detected obstacles. B_p can expand its position dimension given a motion and A_p considers the probability of being at a given state given the position. When we predict a new position given no obstacle in a direction, B_p expands. If the expected motion leads to an un-visited location, s_{t+1} does not exist in the model. The state is undefined while the position p_{t+1} is certain, therefore all Markov matrices are expected to grow in their state dimension to match this new prediction. This process enables the dynamic expansion of the dimensions of both the observation and position likelihoods (A_o, A_p) and state transition (B_s) to consider the novel state s_{t+1} in a process equivalent to [9]. Subsequently, the state transition probability (B_s) and position likelihood (A_p) can be updated through the same equation 4, here shown with a transition matrix.

$$B_\pi = Q(s_{t+1}|s_t, \pi)Q(s_t) * B_\pi * learning\_rate \tag{4}$$

With s_{t+1} being a new state if there are no obstacles detected and a new position is predicted or the same state s_t otherwise. The learning rate is set higher for

experimented transitions than imagined transitions such that we form new connections weaker toward expected places compared to visited places as we would expect from animal synaptic learning [6]. A_o has grown in its state dimension, however, it lacks information regarding the specific observation o_{t+1} expected in that location, resulting in a uniform distribution for that state. Such areas exhibit high uncertainty in their observation likelihood model. Using those prior, the agent can leverage the Active Inference scheme to determine where to direct itself to maximise its objective (e.g. forming a comprehensive map of the environment). While previous models such as [9,29] adjust their internal model growth to accommodate new patterns of observations, we extend the concept to predict, un-visited, areas and generate new states holding no observation. Those unknown states are therefore highly attractive when seeking information gain and largely improve exploration strategy.

3.4 Policy Selection in Active Inference

Policy selection plays a crucial role in exploring those expected states generated by the model. The AIF guides the agent's decision-making process based on the minimisation of expected surprise and uncertainty. Policy selection, informed by the AIF, determines the agent's actions and map extension in response to sensory inputs and internal beliefs.

Typically, agents are assumed to desire to minimise their variational free energy (F), which can serve as a metric to quantify the discrepancy between the joint distribution P and the approximate posterior Q as presented in Equation 5.

$$\begin{aligned} F &= \mathbb{E}_{Q(\tilde{s},\tilde{p}|\tilde{a},\tilde{o})}[\log[Q(\tilde{s},\tilde{p}|\tilde{a},\tilde{o})] - \log[P(\tilde{s},\tilde{p},\tilde{a},\tilde{o})] \\ &= \underbrace{D_{KL}[Q(\tilde{s},\tilde{p}|\tilde{a},\tilde{o}))||P(\tilde{s},\tilde{p}|\tilde{a},\tilde{o})]}_{\text{posterior approximation}} - \underbrace{\log[P(\tilde{o})]}_{\text{log evidence}} \\ &= \underbrace{D_{KL}[Q(\tilde{s},\tilde{p}|\tilde{a},\tilde{o}))||P(\tilde{s},\tilde{p},\tilde{a})]}_{\text{complexity}} - \underbrace{\mathbb{E}_{Q(\tilde{s},\tilde{p}|\tilde{a},\tilde{o})}[\log[P(\tilde{o}|\tilde{s})]}_{\text{accuracy}} \end{aligned} \tag{5}$$

Active inference agents aim to minimise their free energy by engaging in three main processes: learning, perception, and planning. Learning involves optimising the model parameters, perception entails estimating the most likely state, and planning involves selecting the policy or action sequence that leads to the lowest expected free energy. Essentially, this means that the process involves forming beliefs about hidden states that offer a precise and concise explanation of observed outcomes while minimising complexity.

While planning, however, we use the expected free energy (G), indicating the agent's anticipated variational free energy following the implementation of a policy π. Unlike the variational free energy, which focuses on current and past observations, the expected free energy incorporates future expected observations generated by the selected policy.

$$G(\pi,\tau) = \underbrace{\mathbb{E}_{Q(o_\tau,s_\tau|\pi)}[log(Q(s_\tau|\pi) - log(Q(s_\tau|o_\tau,\pi))]}_{\text{information gain term}} - \underbrace{\mathbb{E}_{Q(o_\tau,s_\tau|\pi))}[log(P(o_\tau))]}_{\text{utility term}} \tag{6}$$

The expected information gain quantifies the anticipated shift in the agent's belief over the state from the prior $Q(s_\tau|\pi)$ to the posterior $Q(s_\tau|o_\tau,\pi)$ when pursuing a particular policy. On the other hand, the utility term assesses the expected log probability of observing the preferred outcome under the chosen policy. This value intuitively measures the likelihood that the policy will guide the agent toward its prior preferences. In this study, we give no prior preference to the agent, as it does not know the environment (unknown observations and map size).

To calculate this expected free energy $G(\pi)$ over each step τ of a policy we sum the expected free energy of each time-step.

$$G(\pi) = \sum_{\tau} G(\pi,\tau) \tag{7}$$

To consider the best policy, we recall that active inference achieves goal-directed behaviour by selecting policies minimising this expected free energy, thereby aiming to produce observations closer to preferred outcomes or prior preferences. This is achieved by setting the approximate posterior over policies as in Equation 8 [19]:

$$P(\pi) = \sigma(-\gamma G(\pi)) \tag{8}$$

Where σ, the softmax function is tempered with a temperature parameter γ, given as a hyper-parameter, converting the expected free energy of policies into a categorical distribution over policies. Actions are then sampled based on this posterior distribution, with lower temperatures resulting in more deterministic behaviour.

By navigating without a clear preference, we desire the highest information gain, effectively pushing the agent toward states it anticipates but doesn't know what to expect from.

4 Results

We explore experimental scenarios where an agent navigates within a grid environment with cardinal motions and still motion. The agents have no direct access to a map of the environment and visual observations are considered to undergo hierarchical processing, transforming them from a vector to a single descriptor corresponding to one colour per room. They receive localised sensory inputs, corresponding to the current room they are in. Sensory inputs which are possibly repeated at different locations (aliased observations). Given a series of

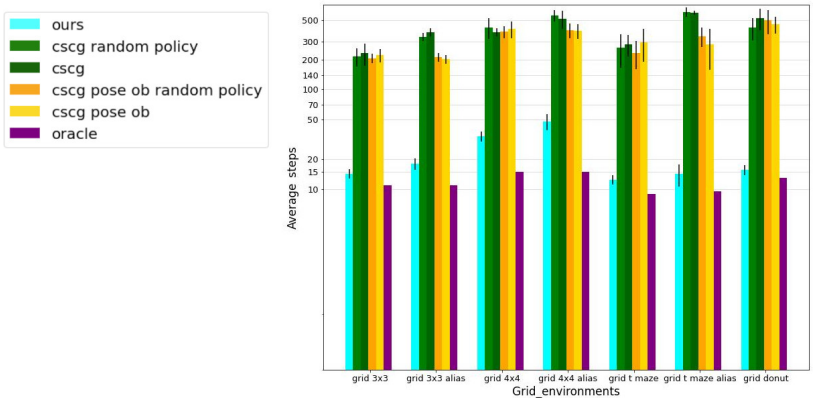
discretised egocentric observations and actions, the agent must deduce the latent topology of its environment to assess various navigation options. Learning this latent graph from aliased observations presents a challenge for most artificial agents [18]. We contrast our model with CSCG [10], a specialised variant of Hidden Markov Models (HMM). CSCG employs a probabilistic approach, using sequences of action-observation pairs without assuming Euclidean geometry. Each observation corresponds to a subset of hidden states known as clones. Although these states share the same observation likelihood, they differ in their implied dynamics encoded in the transition model. By analysing the sequence of action-observation pairs, specific clones with higher likelihoods can disambiguate the aliased observations. Initially, CSCG gathers a dataset through maze exploration to learn the spatial structure [10].

To make the two models more similar for a fair comparison, we include our model's current state estimation mechanism in the CSCG approach [19]. Moreover, we decided to see how CSCG would behave if we included the position as an observation. This effectively removes aliasing and is believed equivalent to the proprioception of our model when starting without prior, we call that specific case "CSCG pose ob". We compared the performance of our model (receiving only observation as input) to the CSCG receiving only visual observations or visual observation-position pairs with or without random exploration policies. The CSCG internal path estimator is based on the Viterbi method [16,10] and is updated every 5 steps with the sequence of pairs going from the first observation to the current time-step.

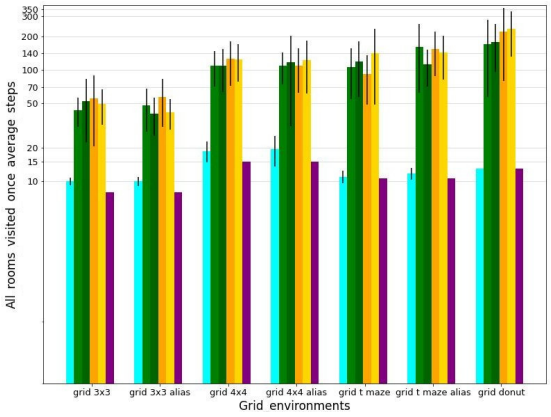
Our environments are composed of several rooms connected in diverse ways (fully connected 3 by 3 and 4 by 4 rooms environments, T-shaped, donuts-shaped mazes with and without aliased floor colours). All models receive the room floor colour as observation. An example of a 3 by 3 rooms environment and extracted observations per room are presented in Figure 1 first and second panel. The agents can move in the four cardinal directions or choose to stay at the present location.

In our exploration runs, across all environments, agents are initially placed at random starting positions and tasked with learning the environment's topology. Results represent the mean over a minimum of ten successful runs in each environment. Notably, our agent always achieves successful exploration, while CSCG occasionally fails due to insufficient steps allotted to learn the topology. The Oracle model, analogous to an A-star path planning, demonstrates the ideal scenario where the agent seizes the full topology of the environment by visiting each position only once. In the case of the T-maze, results are averaged across all runs considering the starting positions of the models.

Our exploration results can be seen in Figure 3a. Exploration is deemed complete when the internal belief over the transition between observations aligns with the ground-truth transition matrix with a minimum certainty of 60% overall correct transitions, the threshold was set arbitrarily based on the resulting successful transition representation as the one that can be seen in a 3 by 3 observations map depicted in figure 4. The figure shows how well-defined are possible



(a) average steps to explore the environments



(b) average steps to discover the environments

Fig. 3. The average steps are depicted on a logarithmic scale. Remarkably, our agent achieves all tasks in significantly fewer steps compared to the CSCG model. The oracle sets the benchmark, representing the minimum steps necessary to visit all rooms once. Additionally, an aliased room signifies the recurrence of identical observations across various locations, posing a challenge as it could mislead the agent regarding its current position.

transitions compared to impossible transitions (due to walls). We also see that giving unique observations (visual observation-position pair) information reduces the CSCG training time of about 100 steps in aliased squared environments and 200 in T-shaped mazes, most probably due to its structure, the agent stays stuck in an aisle. However using random policy or the Viterbi algorithm for navigation does not improve exploration, because the agent can not extrapolate on unseen observations, thus finally leading to almost random action selection. This demonstrates the benefit of map extension over un-visited areas.

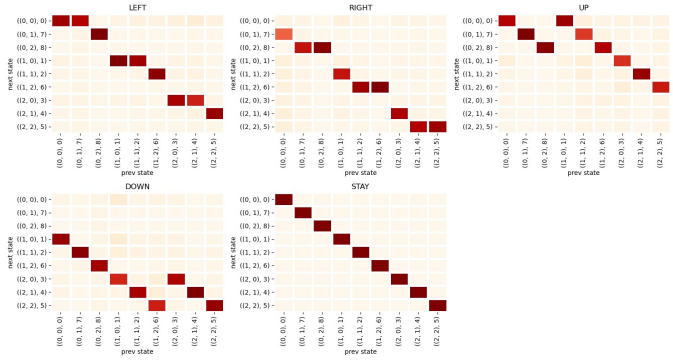
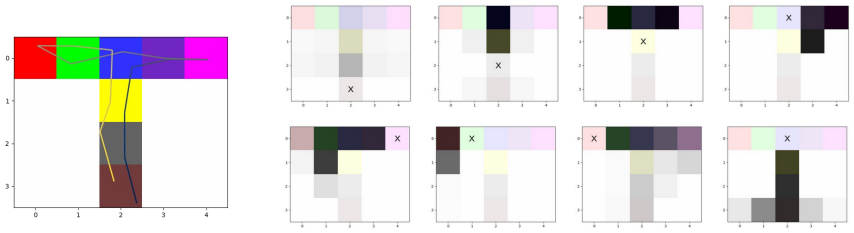


Fig. 4. Example of a successful transition representation between positions in a 3x3 grid map. Each state in the plot is paired with its corresponding ground-truth pose for clarity (pose, state). The intensity of colour in the figure indicates the level of certainty the agent has about the transition.

If we compare the number of steps required to learn the structure of the environment and the number of steps the agent takes to visit all unknown positions, we can deduce a few things. Firstly, not having an imagination over possible trajectories disadvantages the CSCG, as it repeatedly visits known rooms, randomly or not, instead of being attracted to novelty as ours is. Ours has prior over non-visited states, rendering unknown rooms highly uncertain, and thus attractive to diminish the agent's internal model's parameters uncertainty. Secondly, we see our agent exploring all rooms with steps closely matching the oracle, this implies that it could have the potential to learn transitions faster, in a one-shot learning if we were to increase confidence in imagined beliefs. However, this could also consolidate misbeliefs about transitions, in those experiments we let the agent confirm its priors instead of over-trusting them by setting the learning rate of the model low on predicted transitions. The given exploration seems to follow biological evidence on mouse behaviour in a maze [26].

We give a qualitative example of the agent behaviour in the T-maze Figure 5. Figure 5a shows the full path taken by a line varying from black to yellow, the agent starts at the bottom of the T-maze. Figure 5b shows the imagined trajectories of the agent (represented by X in the figure) at various steps to read from left to right and top to bottom. Imagined trajectories are associated with their expected free energy, the darker, the more desirable the path to the agent. The agent is purely driven by information gain in those experiments. Our model has low interest in paths leading into the current room walls and is highly attracted to unexplored areas. Upon reaching the end of the right aisle (1st image, 2^{nd} row of Fig 5b), the unexplored aisle is notably more attractive than the previously visited one, highlighting the agent's preference for uncertain observations over confirming existing beliefs. While returning to the starting point, the agent shows interest in paths going through walls. This is because these transitions become more intriguing as the agent has gained a better understanding of the

environment's connectivity. A consolidation of its belief can be realised through a new observation of the walls. Those observations confirm that the agent exhibits a coherent and effective exploration behaviour akin to how we would explore an environment, first discovering all areas before delving into specific details.



(a) (b)

Fig. 5. Exploration of a T-maze starting at the base of the T. a) depicts the full path as a line transitioning from black to white. b) showcases, from top to bottom columns one to two, the agent -represented as an X- imagined optimal policies. Darker colours indicate higher expected free energy.

5 Discussion

This study proposes a novel high-level abstraction model informed by biologically plausible principles mimicking key points of animal navigation strategies [33,1]. By integrating a dynamic cognitive graph with internal positioning and an Active Inference Framework, our model successfully explores the environment and learns its structure in a few steps, as expected from animals [32,26], facilitating adaptive learning and efficient exploration. Moreover, allowing the internal map to grow with expected beliefs not only creates a map adapted to any environment dimension, shape or observations but also enhances exploration by creating highly uncertain states where the whereabouts are predictable but the corresponding observations aren't. Comparative experiments with the Clone-Structured Graph (CSCG) model [10] underscore the effectiveness of our approach in learning environment structures with minimal data and without prior knowledge of specific observation dimensions. This is mainly due to our agent's capacity to imagine actions' consequences and integrate them into its beliefs. Moving forward, it would be interesting to increase the prediction range of new states to integrate into the model and determine the impact on navigation. Moreover, studying the impact of a perfect memory on future policies and exploration efficiency, as well as seeing how the agent fares when trying to reach a defined objective it has prior upon in a familiar or novel environment would enhance the research. Finally deploying this model in real-world scenarios such as StreetLearn [20], based on Google map observations, would approach further this mechanism to animal behaviour and provide more conclusive evidence.

Acknowledgement

This research received funding from the Flemish Government under the "Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen" programme.

References

1. Balaguer, J., Spiers, H., Hassabis, D., Summerfield, C.: Neural mechanisms of hierarchical planning in a virtual subway network. Neuron **90**, 893–903 (05 2016). https://doi.org/10.1016/j.neuron.2016.03.037
2. Bush, D., Barry, C., Manson, D., Burgess, N.: Using grid cells for navigation. Neuron **87**, 507 – 520 (2015), https://api.semanticscholar.org/CorpusID:7275119
3. Chevalier-Boisvert, M., Willems, L., Pal, S.: Minimalistic gridworld environment for openai gym. https://github.com/maximecb/gym-minigrid (2018)
4. Dedieu, A., Lehrach, W., Zhou, G., George, D., Lázaro-Gredilla, M.: Learning cognitive maps from transformer representations for efficient planning in partially observed environments (2024)
5. Edvardsen, V., Bicanski, A., Burgess, N.: Navigating with grid and place cells in cluttered environments. Hippocampus **30** (08 2019). https://doi.org/10.1002/hipo.23147
6. Eichenbaum, H.: The hippocampus as a cognitive map ... of social space. Neuron **87**(1), 9–11 (2015). https://doi.org/https://doi.org/10.1016/j.neuron.2015.06.013, https://www.sciencedirect.com/science/article/pii/S0896627315005267
7. Epstein, R., Patai, E.Z., Julian, J., Spiers, H.: The cognitive map in humans: Spatial navigation and beyond. Nature Neuroscience **20**, 1504–1513 (10 2017). https://doi.org/10.1038/nn.4656
8. Friston, K., FitzGerald, T., Rigoli, F., Schwartenbeck, P., Doherty, J.O., Pezzulo, G.: Active inference and learning. Neuroscience & Biobehavioral Reviews **68**, 862–879 (2016). https://doi.org/https://doi.org/10.1016/j.neubiorev.2016.06.022, https://www.sciencedirect.com/science/article/pii/S0149763416301336
9. Friston, K., Parr, T., Zeidman, P.: Bayesian model reduction (2019)
10. George, D., Rikhye, R., Gothoskar, N., Guntupalli, J.S., Dedieu, A., Lázaro-Gredilla, M.: Clone-structured graph representations enable flexible learning and vicarious evaluation of cognitive maps. Nature Communications **12** (04 2021). https://doi.org/10.1038/s41467-021-22559-5
11. Gershman, S.: Deconstructing the human algorithms for exploration. Cognition **173**, 34–42 (12 2017). https://doi.org/10.1016/j.cognition.2017.12.014
12. Guntupalli, J.S., Raju, R., Kushagra, S., Wendelken, C., Sawyer, D., Deshpande, I., Zhou, G., Lázaro-Gredilla, M., George, D.: Graph schemas as abstractions for transfer learning, inference, and planning (02 2023). https://doi.org/10.48550/arXiv.2302.07350
13. Heins, R.C., Mirza, M.B., Parr, T., Friston, K., Kagan, I., Pooresmaeili, A.: Deep active inference and scene construction. Frontiers in Artificial Intelligence **3** (2020). https://doi.org/10.3389/frai.2020.509354, https://www.frontiersin.org/articles/10.3389/frai.2020.509354
14. Hinman, J., Brandon, M., Climer, J., Chapman, W., Hasselmo, M.: Multiple running speed signals in medial entorhinal cortex. Neuron **91** (07 2016). https://doi.org/10.1016/j.neuron.2016.06.027

15. Jacobs, J., Kahana, M., Ekstrom, A., Mollison, M., Fried, I.: A sense of direction in human entorhinal cortex. Proceedings of the National Academy of Sciences of the United States of America **107**, 6487–92 (03 2010). https://doi.org/10.1073/pnas.0911213107
16. Jelinek, F.: Continuous speech recognition by statistical methods. Proceedings of the IEEE **64**(4), 532–556 (1976). https://doi.org/10.1109/PROC.1976.10159
17. Kaplan, R., Friston, K.: Planning and navigation as active inference (12 2017). https://doi.org/10.1101/230599
18. Lajoie, P., Hu, S., Beltrame, G., Carlone, L.: Modeling perceptual aliasing in SLAM via discrete-continuous graphical models. CoRR **abs/1810.11692** (2018), http://arxiv.org/abs/1810.11692
19. de Maele, T.V., Dhoedt, B., Verbelen, T., Pezzulo, G.: Bridging cognitive maps: a hierarchical active inference model of spatial alternation tasks and the hippocampal-prefrontal circuit (2023)
20. Mirowski, P., Banki-Horvath, A., Anderson, K., Teplyashin, D., Hermann, K.M., Malinowski, M., Grimes, M.K., Simonyan, K., Kavukcuoglu, K., Zisserman, A., Hadsell, R.: The streetlearn environment and dataset. CoRR **abs/1903.01292** (2019), http://arxiv.org/abs/1903.01292
21. Neacsu, V., Mirza, M.B., Adams, R.A., Friston, K.J.: Structure learning enhances concept formation in synthetic active inference agents. PLOS ONE **17**(11), 1–34 (11 2022). https://doi.org/10.1371/journal.pone.0277199, https://doi.org/10.1371/journal.pone.0277199
22. Parr, T., Pezzulo, G., Friston, K.: Active Inference: The Free Energy Principle in Mind, Brain, and Behavior (03 2022). https://doi.org/10.7551/mitpress/12441.001.0001
23. Peer, M., Brunec, I.K., Newcombe, N.S., Epstein, R.A.: Structuring knowledge with cognitive maps and cognitive graphs. Trends in Cognitive Sciences **25**(1), 37–54 (2021). https://doi.org/https://doi.org/10.1016/j.tics.2020.10.004, https://www.sciencedirect.com/science/article/pii/S1364661320302503
24. Peer, M., Brunec, I.K., Newcombe, N.S., Epstein, R.A.: Structuring knowledge with cognitive maps and cognitive graphs. Trends in Cognitive Sciences **25**(1), 37–54 (2021). https://doi.org/https://doi.org/10.1016/j.tics.2020.10.004, https://www.sciencedirect.com/science/article/pii/S1364661320302503
25. Raju, R.V., Guntupalli, J.S., Zhou, G., Lázaro-Gredilla, M., George, D.: Space is a latent sequence: Structured sequence learning as a unified theory of representation in the hippocampus (2022)
26. Rosenberg, M., Zhang, T., Perona, P., Meister, M.: Mice in a labyrinth show rapid learning, sudden insight, and efficient exploration. eLife **10**, e66175 (jul 2021). https://doi.org/10.7554/eLife.66175, https://doi.org/10.7554/eLife.66175
27. Schwartenbeck, P., Passecker, J., Hauser, T.U., FitzGerald, T.H., Kronbichler, M., Friston, K.J.: Computational mechanisms of curiosity and goal-directed exploration. eLife **8**, e41703 (may 2019). https://doi.org/10.7554/eLife.41703, https://doi.org/10.7554/eLife.41703
28. Smith, R., Friston, K.J., Whyte, C.J.: A step-by-step tutorial on active inference and its application to empirical data. Journal of Mathematical Psychology **107**, 102632 (2022). https://doi.org/https://doi.org/10.1016/j.jmp.2021.102632, https://www.sciencedirect.com/science/article/pii/S0022249621000973
29. Smith, R., Schwartenbeck, P., Parr, T., Friston, K.J.: An active inference approach to modeling structure learning: Concept learning as an example case. Frontiers in Computational

Neuroscience **14** (2020). https://doi.org/10.3389/fncom.2020.00041, https://www.frontiersin.org/articles/10.3389/fncom.2020.00041
30. Solstad, T., Boccara, C.N., Kropff, E., Moser, M.B., Moser, E.I.: Representation of geometric borders in the entorhinal cortex. Science **322**(5909), 1865–1868 (2008). https://doi.org/10.1126/science.1166466, https://www.science.org/doi/abs/10.1126/science.1166466
31. de Tinguy, D., Van de Maele, T., Verbelen, T., Dhoedt, B.: Spatial and temporal hierarchy for autonomous navigation using active inference in minigrid environment. Entropy **26**(1), 83 (Jan 2024). https://doi.org/10.3390/e26010083, http://dx.doi.org/10.3390/e26010083
32. Tyukin, I.Y., Gorban, A.N., Alkhudaydi, M.H., Zhou, Q.: Demystification of few-shot and one-shot learning. CoRR **abs/2104.12174** (2021), https://arxiv.org/abs/2104.12174
33. Zhao, M.: Human spatial representation: What we cannot learn from the studies of rodent navigation. Journal of Neurophysiology **120** (08 2018). https://doi.org/10.1152/jn.00781.2017

Coupled autoregressive active inference agents for control of multi-joint dynamical systems

Tim N. Nisslbeck[1 [0009−0007−3114−812X]] and
Wouter M. Kouw[1 [0000−0002−0547−4817]]

Bayesian Intelligent Autonomous Systems lab, TU Eindhoven, Netherlands
t.n.nisslbeck@tue.nl(✉), w.m.kouw@tue.nl

Abstract. We propose an active inference agent to identify and control a mechanical system with multiple bodies connected by joints. This agent is constructed from multiple scalar autoregressive model-based agents, coupled together by virtue of sharing memories. Each subagent infers parameters through Bayesian filtering and controls by minimizing expected free energy over a finite time horizon. We demonstrate that a coupled agent of this kind is able to learn the dynamics of a double mass-spring-damper system, and drive it to a desired position through a balance of explorative and exploitative actions. It outperforms the uncoupled subagents in terms of surprise and goal alignment.

Keywords: Active inference · Expected free energy minimization · Autoregressive models · Bayesian filtering · Adaptive control.

1 Introduction

Our society relies heavily on mechatronic systems for manufacturing, energy, transport, logistics and healthcare. These systems are still largely designed using physics-driven models, offline system identification and optimal control techniques. However, this design framework leads to systems that tend to be sensitive to "noise", i.e., sensor and actuator imperfections, external disturbances, and unmodeled physics (e.g., heat, vibrations). Robustness requires adaptation to a changing environment by updating a model rapidly, continuously and data-efficiently. This is exactly what embodied artificial intelligence and cognitive robotics strive to achieve [17,14]. Reinforcement learning is a prime candidate framework, but it tends to be costly in terms of computational resources and training time [3]. A more appropriate framework for resource-constrained mechatronic systems is active inference, which characterizes itself by including optimal information gain in its data acquisition protocol [22,21]. Here we present scalar active inference agents that are coupled together to jointly control a mechatronic system with multiple inputs and multiple outputs [20].

Active inference draws its roots from cognitive science where it is a process theory for intelligent behaviour [8]. Many agents with discrete state and action spaces have been proposed as models of learning, exploration and curiosity [7,5,26,4]. The engineering community wants to use active inference as a

framework for designing intelligent autonomous systems with continuous state and action spaces [23,16,1,2,12]. A major challenge in designing such agents are the calculations of the differential entropies involved. Many models assume some form of Gaussian state transition or likelihood, often with parameters shaped by neural networks [28,10,11]. We build on recent work using autoregressive models fit for resource-constrained mechatronic systems [13]. Our contributions include:

- The formulation of a coupled active inference agent consisting of two scalar agents that share memories (Sec. 3.3).
- An empirical evaluation of coupled versus uncoupled agents on a double mass-spring-damper system (Sec. 4).

2 Problem statement

We study the class of multi-joint dynamical systems, characterized by simple mechanical systems connected in sequence. For example, a double mass-spring-damper system consists of one mass attached to a base through a spring and an accompanying damper, with a second mass connected to the first mass through another spring and damper (Figure 1 left). Similarly, a double pendulum consists of a single pendulum attached to a base and another single pendulum attached to the end of the first pendulum (Figure 1 right). The task is to find control policies for each motor such that the multi-joint dynamical system moves to a desired position. We expect that coupling agents together lets them more accurately predict joint motion and infer an appropriate control policy sooner.

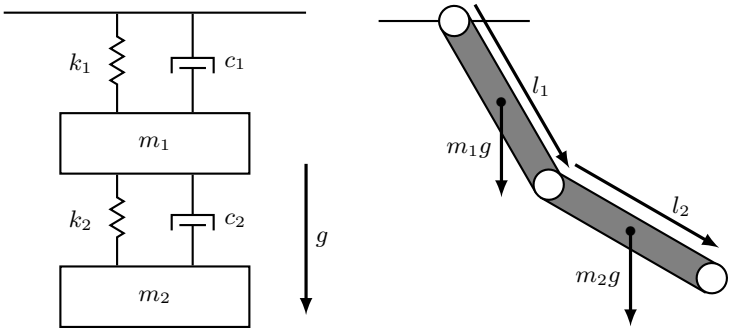


Fig. 1: (Left) A double mass-spring-damper system where block 1 is attached to a stationary frame and block 2 is attached to the first block. The dynamics of the system are determined by the masses m_i of the blocks, the stiffness of the springs k_i, the amount of friction c_i the dampeners provide and gravity g. (Right) A double compound pendulum system consisting of two single compound pendulums joined end-to-end. The dynamics of the system are determined by the masses m_i and lengths l_i of the poles.

3 Agent specification

Consider an agent, operating in discrete time, that sends inputs $u_k \in \mathbb{R}$ (a.k.a. controls, actions) to a system and measures its output $y_k \in \mathbb{R}$. The agent must drive the system to a desired output y_* without knowledge of its dynamics. Since this active inference agent minimizes expected free energy (EFE) based on an autoregressive exogenous (ARX) model, we refer to it as an ARX-EFE agent.

3.1 Probabilistic model

We specify a likelihood function of the form:

$$p(y_k \mid \theta, \tau, u_k, \bar{u}_k, \bar{y}_k) = \mathcal{N}\big(y_k \mid \theta^\intercal \begin{bmatrix} u_k & \bar{u}_k & \bar{y}_k \end{bmatrix}, \tau^{-1}\big)\,, \tag{1}$$

where the vectors $\bar{y}_k \in \mathbb{R}^{M_y}$ and $\bar{u}_k \in \mathbb{R}^{M_u}$ are buffers containing previous observations of the system outputs and inputs, where M_y and M_u are the lengths of the output and input buffers, respectively. This defines the above likelihood as an autoregressive model. $\theta \in \mathbb{R}^D$, where $D = M_y + M_u + 1$, are coefficients and $\tau \in \mathbb{R}^+$ represents a precision parameter.

The prior distribution on the parameters is a multivariate Gaussian - univariate Gamma distribution [27, ID: D5]:

$$p(\theta, \tau) \triangleq \mathcal{NG}\big(\theta, \tau \mid \mu_0, \Lambda_0, \alpha_0, \beta_0\big) = \mathcal{N}(\theta \mid \mu_0, (\tau\Lambda_0)^{-1})\,\mathcal{G}\big(\tau \mid \alpha_0, \beta_0\big)\,. \tag{2}$$

The prior distributions over inputs are assumed to be independent over time:

$$p(u_k) \triangleq \mathcal{N}(u_k \mid 0, \eta^{-1})\,, \tag{3}$$

with precision parameter η. This choice has a regularizing effect on the inferred controls (Sec. 3.2).

3.2 Inference

Our inference procedure is separated into a parameter belief update procedure given observed data, and control estimation given parameters.

Parameters First, note that, at time k, the control u_k has been executed and is known to the agent. Henceforth, we shall use $\hat{u}_k$ and $\hat{y}_k$ to differentiate observed variables from unobserved ones. Furthermore, let

$$x_k = \begin{bmatrix} u_k & \bar{u}_k & \bar{y}_k \end{bmatrix}\,. \tag{4}$$

The parameter posterior distribution is obtained by Bayesian filtering [25]:

$$\underbrace{p\big(\theta, \tau \mid \mathcal{D}_k\big)}_{\text{posterior}} = \frac{\overbrace{p\big(\hat{y}_k \mid \theta, \tau, \hat{u}_k, \bar{u}_k, \bar{y}_k\big)}^{\text{likelihood}}}{\underbrace{p\big(\hat{y}_k \mid \hat{u}_k, \mathcal{D}_{k\text{-}1}\big)}_{\text{evidence}}}\,\underbrace{p\big(\theta, \tau \mid \mathcal{D}_{k\text{-}1}\big)}_{\text{prior}}\,. \tag{5}$$

where $\mathcal{D}_k = \{\hat{y}_i, \hat{u}_i\}_{i=1}^k$ is the data up to time k. The evidence (a.k.a. marginal likelihood) is

$$p\big(\hat{y}_k \mid \hat{u}_k, \mathcal{D}_{k\text{-}1}\big) = \int p\big(\hat{y}_k \mid \theta, \tau, \hat{u}_k, \bar{u}_k, \bar{y}_k\big) p\big(\theta, \tau \mid \mathcal{D}_{k\text{-}1}\big)\, \mathrm{d}(\theta, \tau). \tag{6}$$

We obtain an exact posterior distribution using the multivariate Gaussian - univariate Gamma prior distribution specified in Eq. 2 [13]:

$$p(\theta, \tau \mid \mathcal{D}_k) = \mathcal{NG}(\theta, \tau \mid \mu_k, \Lambda_k, \alpha_k, \beta_k)\,. \tag{7}$$

where

$$\mu_k = \big(x_k x_k^\intercal + \Lambda_{k\text{-}1}\big)^{-1}\big(x_k \hat{y}_k + \Lambda_{k\text{-}1}\mu_{k\text{-}1}\big)\,, \quad \Lambda_k = x_k x_k^\intercal + \Lambda_{k\text{-}1}, \tag{8}$$

$$\alpha_k = \alpha_{k\text{-}1} + \frac{1}{2}\,, \quad \beta_k = \beta_{k\text{-}1} + \frac{1}{2}\big(\hat{y}_k^2 - \mu_k^\intercal \Lambda_k \mu_k + \mu_{k\text{-}1}^\intercal \Lambda_{k\text{-}1}\mu_{k\text{-}1}\big)\,. \tag{9}$$

The marginal posterior distributions are Gamma distributed and multivariate location-scale T-distributed [27, ID: P36]:

$$p(\tau \mid \mathcal{D}_k) = \int p(\theta, \tau \mid \mathcal{D}_k)\mathrm{d}\theta = \mathcal{G}(\tau \mid \alpha_k, \beta_k)\,, \tag{10}$$

$$p(\theta \mid \mathcal{D}_k) = \int p(\theta, \tau \mid \mathcal{D}_k)\mathrm{d}\tau = \mathcal{T}_{2\alpha_k}\big(\theta \mid \mu_k, \frac{\beta_k}{\alpha_k}\Lambda_k^{\text{-}1}\big). \tag{11}$$

The $2\alpha_k$ subscript refers to the T-distribution's degrees of freedom parameter.

Controls In order to effectively drive the system to the goal, the agent must make accurate predictions for future outputs. The predictive probability of the input, output and parameters at time $t = k + 1$ is:

$$p(y_t, \theta, \tau, u_t \mid \mathcal{D}_k) = p(y_t \mid \theta, \tau, u_t, \bar{u}_t, \bar{y}_t)\, p(\theta, \tau \mid \mathcal{D}_k) p(u_t)\,. \tag{12}$$

Note that at time $t = k{+}1$, the buffers $\bar{y}_t = [\hat{y}_k\ \hat{y}_{k-1}\ \ldots]$ and $\bar{u}_t = [\hat{u}_k\ \hat{u}_{k-1}\ \ldots]$ contain only observed variables (i.e., there are no products between random variables). To incorporate the goal output, we invert (see Eq. 21) the conditional dependency in the predictive probability for the output and parameters:

$$p(y_t \mid \theta, \tau, u_t, \bar{u}_t, \bar{y}_t)\, p(\theta, \tau \mid \mathcal{D}_k) = p(y_t, \theta, \tau \mid u_t; \mathcal{D}_k) \tag{13}$$

$$= p(\theta, \tau \mid y_t, u_t; \mathcal{D}_k)\, p(y_t)\,. \tag{14}$$

We intervene on the marginal prior distribution over future output, $p(y_t)$, with our chosen goal prior parameters:

$$p(y_t) \to p(y_t \mid y_*) \triangleq \mathcal{N}(y_t \mid m_*, v_*)\,. \tag{15}$$

Now, to infer a posterior distribution for the control variable u_t, we introduce an expected free energy functional [7,15],

$$\mathcal{F}_k[q] \triangleq \mathbb{E}_{q(y_t, \theta, \tau, u_t)}\Big[\ln \frac{p(\theta, \tau \mid \mathcal{D}_k) q(u_t)}{p(\theta, \tau \mid y_t, u_t; \mathcal{D}_k) p(y_t \mid y_*) p(u_t)}\Big], \tag{16}$$

with a variational model of the form:

$$q(y_t, \theta, \tau, u_t) \triangleq p(y_t, \theta, \tau \mid u_t; \mathcal{D}_k) q(u_t)\,. \tag{17}$$

Inferring the optimal control at time t refers to minimizing the free energy functional with respect to the variational distribution $q(u_t)$:

$$q^*(u_t) = \underset{q \in Q}{\arg\min}\ \mathcal{F}_k[q]\,. \tag{18}$$

where Q represents the space of candidate distributions. We can re-arrange the free energy functional to simplify the variational minimization problem:

$$\mathbb{E}_{q(y_t,u_t,\theta,\tau)}\Big[\ln \frac{p(\theta,\tau \mid \mathcal{D}_k)\, q(u_t)}{p(\theta,\tau \mid y_t, u_t; \mathcal{D}_k) p(y_t \mid y_*) p(u_t)}\Big] = \tag{19}$$
$$\mathbb{E}_{q(u_t)}\Big[\underbrace{\mathbb{E}_{p(y_t,\theta,\tau \mid u_t;\mathcal{D}_k)}\big[\ln \frac{p(\theta,\tau \mid \mathcal{D}_k)}{p(\theta,\tau \mid y_t, u_t; \mathcal{D}_k) p(y_t|y_*)}\big]}_{\mathcal{J}_k(u_t)} + \ln \frac{q(u_t)}{p(u_t)}\Big].$$

Using $\mathcal{J}_k(u_t) = \ln(1/\exp(-\mathcal{J}_k(u_t)))$, the expected free energy functional can be expressed as a Kullback-Leibler divergence

$$\mathcal{F}_k[q] = \mathbb{E}_{q(u_t)}\Big[\ln \frac{q(u_t)}{\exp\big(-\mathcal{J}_k(u_t)\big) p(u_t)}\Big]\,, \tag{20}$$

which is minimal when $q^*(u_t) = \exp\big(-\mathcal{J}_k(u_t)\big) p(u_t)$ [19]. Thus, we have an optimal approximate posterior distribution over controls.

The only unknown distribution in $\mathcal{J}_k(u_t)$ is the distribution over parameters given the future output and control (see Eq. 13). It can be related to known distributions through Bayes' rule:

$$p(\theta,\tau \mid y_t, u_t; \mathcal{D}_k) = \frac{p(y_t \mid \theta, \tau, u_t, \bar{u}_t, \bar{y}_t)\ p(\theta,\tau \mid \mathcal{D}_k)}{\int p(y_t \mid \theta, \tau, u_t, \bar{u}_t, \bar{y}_t) p(\theta,\tau \mid \mathcal{D}_k) \mathrm{d}(\theta,\tau)}. \tag{21}$$

The distribution that results from the marginalization in the denominator is the posterior predictive distribution $p(y_t \mid u_t; \mathcal{D}_k)$ and can be derived analytically within our model specification [13]:

$$p(y_t \mid u_t; \mathcal{D}_k) \triangleq \int p(y_t \mid \theta, \tau, u_t, \bar{u}_t, \bar{y}_t) p(\theta,\tau \mid \mathcal{D}_k) \mathrm{d}(\theta,\tau) \tag{22}$$
$$= \mathcal{T}_{2\alpha_k}\Big(y_t \mid \mu_k^\intercal x_t,\ \frac{\beta_k}{\alpha_k}\big(x_t^\intercal \Lambda_k^{-1} x_t + 1\big)\Big)\,, \tag{23}$$

for $x_t = \big[u_t\ \bar{u}_t\ \bar{y}_t\big]$. If we replace $p(\theta,\tau|y_t,u_t)$ in the expected free energy function with the right-hand side of Eq. 21 and use Eq. 12, then it can be split into two components:

$$\mathcal{J}_k(u_t) = \mathbb{E}_{p(y_t \mid u_t;\mathcal{D}_k)}\Big[-\ln p(y_t \mid y_*)\Big] \tag{24}$$
$$- \mathbb{E}_{p(y_t,\theta,\tau \mid u_t;\mathcal{D}_k)}\Big[\ln \frac{p(y_t,\theta,\tau \mid u_t;\mathcal{D}_k)}{p(\theta,\tau \mid \mathcal{D}_k) p(y_t \mid u_t; \mathcal{D}_k)}\Big]\,.$$

One may recognize the first term as a cross-entropy, describing the dissimilarity between the posterior predictive distribution and the goal prior distribution [19]. The second term is the mutual information between the parameter posterior and the predictive distribution, describing how much information is gained on the parameters upon measuring a system output [19]. Solving the expectations yields

$$\mathcal{J}_k(u_t) = \mathrm{C} + \frac{1}{2v_*}\left((\mu_k^\intercal x_t - m_*)^2 + \frac{\beta_k}{\alpha_k\text{-}1}(x_t^\intercal \Lambda_k^{-1} x_t + 1)\right) - \frac{1}{2}\ln(x_t^\intercal \Lambda_k^{-1} x_t + 1), \quad (25)$$

where C are constants that do not depend on u_t [13].

Unfortunately, the functional form of $q^*(u_t)$ does not appear to be a member of a known parametric family. This means we do not have access to analytic solutions of the moments of this distribution. If only its most probable value is of interest, then the most straightforward approach is maximum a posteriori (MAP) estimation. The MAP estimator can be written as a minimization over a negative logarithmic transformation of $q^*(u_t)$:

$$\hat{u}_t = \arg\max_{u_t \in \mathcal{U}} q^*(u_t) \quad (26)$$

$$= \arg\min_{u_t \in \mathcal{U}} \mathcal{J}_k(u_t) - \ln p(u_t)\,, \quad (27)$$

where $\mathcal{U} = \{u \in \mathbb{R} \,|\, u_{min} \leq u \leq u_{max}\}$ refers to the space of affordable controls. It can be used to incorporate practical constraints such as torque limits. If an approximate uncertainty over the controls is required, then the above MAP estimate can be extended to a Laplace approximation [6].

3.3 Coupling

The above ARX-EFE agent is scalar and can only operate on single-input single-output systems. We can of course naively group multiple such agents together to operate on a multi-input multi-output system, as is sometimes done with Gaussian processes [29, Sec. 9.1]. But that ignores correlations between outputs which is important for prediction of motion in mechanical systems. We propose to couple agents together by virtue of incorporating additional signals into the autoregressive data buffers (i.e., memories) x_t. For agent $j \neq i$, sharing the output buffer between agents would take the form of:

$$p(y_{i,k} \,|\, \theta_i, \tau_i, u_{i,k}, \bar{u}_{i,k}, \bar{y}_{i,k}, \bar{y}_{j,k}) = \mathcal{N}\left(y_{i,k} \,|\, \theta_i^\intercal \left[u_{i,k}\ \bar{u}_{i,k}\ \bar{y}_{i,k},\ \bar{y}_{j,k}\right], \tau^{-1}\right). \quad (28)$$

Through sharing data buffers, the prediction for one system component will depend explicitly on another component. However, this solution poses a problem for when the agent wants to extend its time horizon to $t > k+1$. In principle, due to the independence assumptions on the prior $p(u_t)$ and the variational control posteriors $q(u_t)$, the joint control posterior distribution can be formed as:

$$q^*(u_t, \ldots, u_{t+T}) = \prod_{t=1}^{T} p(u_t)\, \exp\left(-\mathcal{J}_k(u_t)\right). \quad (29)$$

For a single agent, the output buffer for $t > k+2$ can be filled with the maximum a posteriori value of its prediction at $t = k+1$ [13]. This solution can be applied recursively so the time horizon can be extended arbitrarily far. However, in a coupled setting, agent 1 has to use agent 2's prediction for $k+1$. But agent 2's prediction depends on agent 1's action. Thus, the coupled agents must solve a *nested* optimization procedure, iteratively alternating between two scalar optimization procedures. This means coupling becomes computationally expensive for time horizons $t > k+1$.

3.4 Optimization

The optimization problem in Eq. 27 can be solved in a number of ways. Firstly, using modern automatic or algorithmic differentiation tools, the gradient and Hessian with respect to u_t can be obtained. Iterative procedures such as gradient descent or (quasi-)Newton methods, will then return approximate minimizers. The most straightforward way to enforce control space constraints is to utilize an interior-point method [9]. Such a method imposes a log-barrier function, which increases an objective function drastically as it approaches the constraint boundary.

Alternatively, one could quantize the control space $\mathcal{U}$, calculate Eq. 27 for every possible value and select the minimizer. For a single time-step and a scalar control, this procedure may actually be computationally cheaper as it does not require iteration. It does come at the cost of a quantization error for the estimated $\hat{u}_t$, and, of course, it does not scale well for longer time horizons due to the curse of dimensionality (the discretization interval becomes a tensor).

4 Experiments

4.1 System description

We perform an experiment[1] on a double mass-spring-damper system. Its equation of motions are the following second-order ordinary differential equations (ODE) [18]:

$$\begin{bmatrix} m_1 & 0 \\ 0 & m_2 \end{bmatrix} \begin{bmatrix} \ddot{z}_1 \\ \ddot{z}_2 \end{bmatrix} = \begin{bmatrix} -(c_1+c_2) & c_2 \\ c_2 & -c_2 \end{bmatrix} \begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} + \begin{bmatrix} -(k_1+k_2) & k_2 \\ k_2 & -k_2 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \tag{30}$$

where each block i has displacement (or position) z_i, velocity $\dot{z}_i$, acceleration $\ddot{z}_i$, mass i, damping coefficient c_i, spring coefficient k_i, and external force (control) u_i. In our experiments, we choose $c_1 = c_2 = 0.1$, $k_1 = k_2 = 1.0$, and $m_1 = m_2 = 1.0$. To update the state of the system, we numerically solve the ODE using the second-order Størmer-Verlet integration method [24]. This method involves updating the displacement of the mass as follows:

$$z_{t+1} = z_t + \Delta t \dot{z}_t + \frac{1}{2}\Delta t^2 \ddot{z}_t, \tag{31}$$

[1] Code found at https://github.com/biaslab/IWAI2024-CARXEFE

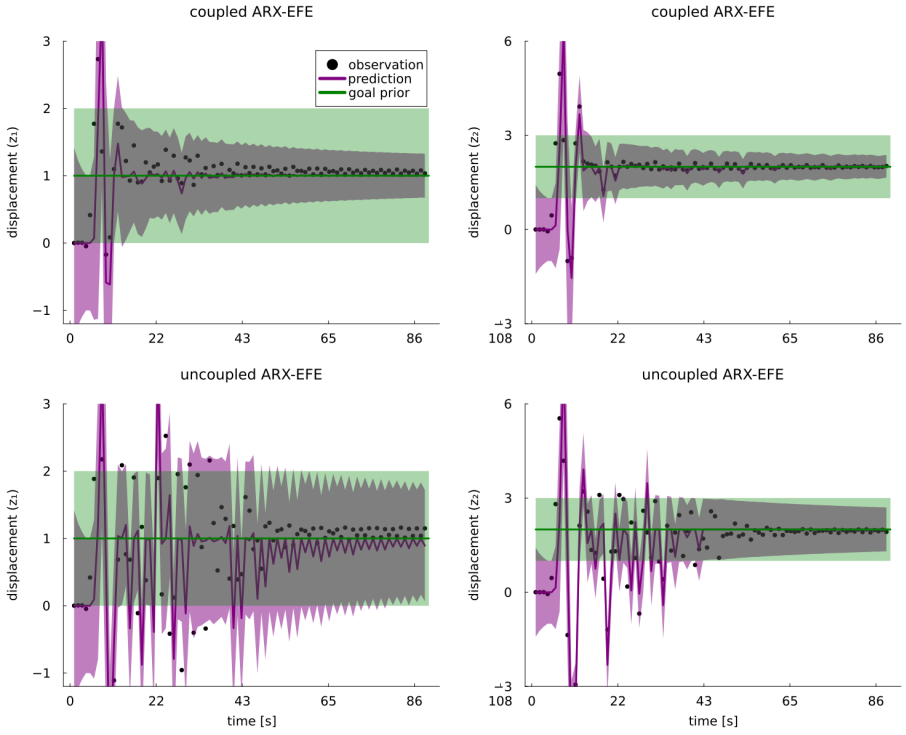
where $\ddot{z}_t$ is calculated from the equations of motion in Eq. 30. The initial state of the system is the fixed point $z_0 = [0.0, 0.0, 0.0, 0.0]$. By reducing the step size Δt and correspondingly increasing the number of updates n_{iter}, we can reduce the risk of numerical instabilities. In our experiments, we choose $\Delta t = 0.01$ and $n_{iter} = 120$. The observed measurement y_t at time t of the system is the position z_t plus measurement noise $\varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2 I_{M_y})$, where I_{M_y} is an identity matrix of size $M_y \times M_y$, and σ_ε^2 is the variance of the noise. We use $\sigma_\varepsilon^2 = 1 \times 10^{-5}$. We discretize the control space $\mathcal{U}$ into $n_\mathcal{U} = 999$ discrete controls, $\mathcal{U} = \{u_{min} + \frac{k(u_{max} - u_{min})}{n_\mathcal{U} - 1} \mid k = 0, 1, 2, \ldots, n_\mathcal{U} - 1\}$, using control limits $u_{min} = -1.0, u_{max} = 1.0$. A multi-joint dynamical system has control space $\mathcal{U}^{D_u}$ and observation space $\mathbb{R}^{D_y}$ with dimensions $D_y > 1$ and $D_u > 1$, respectively. Since we couple ARX-EFE agents with single input and single output, we require $D_y = D_u$ agents to control and observe the system. In the case of a double mass-spring-damper system, $D_y = D_u = 2$.

4.2 Comparisons

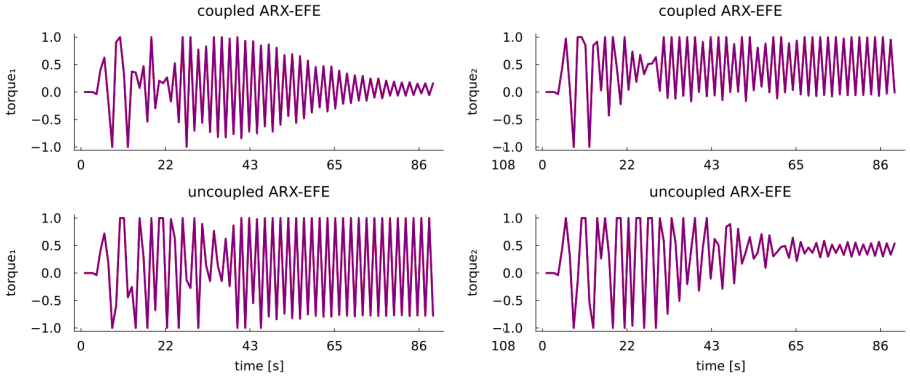
We compare a set of coupled ARX-EFE agents, referred to as CARX-EFE agents, with a set of uncoupled ARX-EFE agents. CARX-EFE and uncoupled ARX-EFE agents differ in the size M of the history vector x_k. For each buffer type, we use a history size of 2. Thus, an uncoupled ARX-EFE agent has a memory size $M = 4$ (2 each for a history of its own observations and controls). CARX-EFE has a memory size of $M = 6$, as we additionally include a history of observations of the other agent. Each agent has a set of parameters $(\mu_0, \Lambda_0, \alpha_0, \beta_0, \eta_0)$. By initializing μ_0 as a zero matrix and Λ_0 as an identity matrix (each of size M), we ensure initial conditions for optimization that give each element in x_t equal importance to calculate the control objective in Eq. 25. We further choose $\alpha_0 = 2.0$, $\beta_0 = 3.0$, and $\eta_0 = 0.001$. The parameters of the goal priors for each agent are $(m_{1,*}, v_{1,*}) = (1.0, 1.0)$ and $(m_{2,*}, v_{2,*}) = (2.0, 1.0)$.

4.3 Results

Figure 2a shows the displacements of the two masses (z_1 for mass m_1 on the left and z_2 for mass m_2 on the right) as a function of time for the coupled agents (top row) compared to the uncoupled agents (bottom row). The black scatter points show the observations that the system generated, while the agent's one-step ahead predictions are shown as purple lines, accompanied by ribbons indicating one standard deviation of the prediction variance. The goal prior, indicating the desired displacement over time, is shown in green with a ribbon reflecting one standard deviation of the goal prior variance. The CARX-EFE agents demonstrate rapid stabilization around the goal prior, with displacements converging towards the goal prior within the first 20 time steps. After reaching the goal prior, oscillations around it diminish over time, resulting in a stable state where both displacements remain within a narrow range of the desired values, as indicated by the low prediction variance. In contrast, the uncoupled



(a) Observations (scatter points) and predictions of displacements of the two blocks (left = displacement z_1 of mass m_1, right = displacement z_2 of mass m_2, in purple), plotted over time. Goal prior distributions plotted in green. Both the prediction and goal prior variance are indicated by a shaded ribbon corresponding to one standard deviation. Compared to its uncoupled counterpart, CARX-EFE achieves lower prediction uncertainty, as indicated by lower prediction variance.



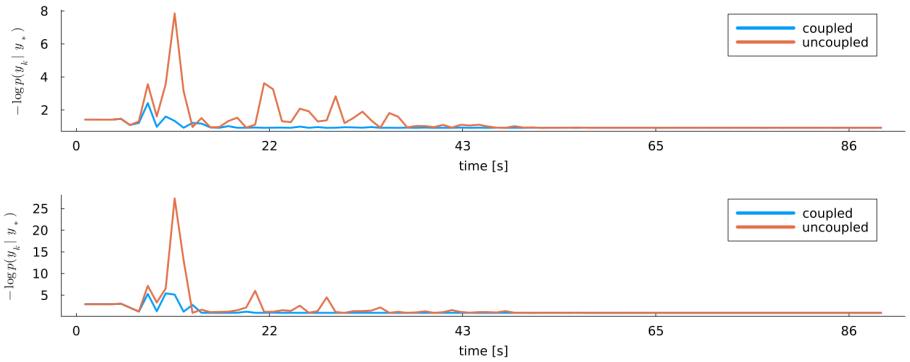
(b) Controls plotted over time. All agents exhibit a short inactivity phase in the beginning, before choosing non-zero controls. The control signals for both the coupled agent controlling mass m_1 and the uncoupled agent controlling mass m_2 have an initial period of large oscillations, which gradually diminish in amplitude, eventually converging to specific values (0.0 for the coupled agent, 0.4 for the uncoupled agent) with narrower oscillations.

Fig. 2: Comparison of predictions and controls of a set of CARX-EFE agents (top rows) versus a set of uncoupled ARX-EFE agents (bottom rows). Each column represents an agent controlling the first and second mass, respectively.

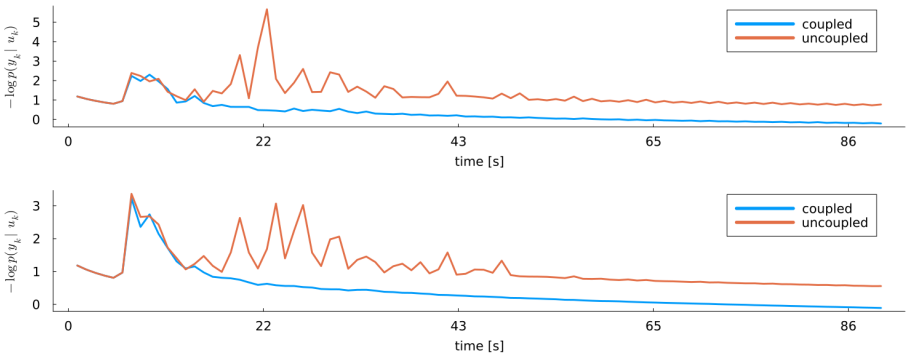
ARX-EFE agents oscillate more wildly (until around time step 45) and have more difficulty maintaining close adherence to the goal prior. They exhibit a prolonged oscillatory phase, where oscillations are more persistent and take significantly longer to dampen. The higher prediction variance further highlights the increased uncertainty and instability in the performance of uncoupled ARX-EFE agents compared to CARX-EFE. The control signals (Fig. 2b) provide further insight into the observed differences in stabilization performance. Both sets of agents start with a brief initial phase of inactivity, during which the control signals remain at zero, keeping the system in its initial, stable state. Following this inactivity phase, both sets of agents apply non-zero control inputs characterized by relatively large oscillations where they learn the input-output relationship before moving to the goal prior. After reaching the goal prior, the control pulse width of one agent in each set gradually converges to specific values (0.0 for the coupled agent, 0.4 for the uncoupled agent), while the other agent in the set alternates between a high and a low control value of the control space $\mathcal{U}$. These oscillations are more narrow for the agent in control of mass m_2, compared to the uncoupled ARX-EFE agent controlling mass m_1.

Figure 3 compares the model performance of both agent sets over time, divided into two subplots: goal alignment (Fig. 3a) and surprise (Fig. 3b). Each subplot is further split into two rows, showing the performance of agents controlling the first and second mass, respectively. Goal alignment, quantified as $-\log p(y_t \mid y_*)$, measures how closely the agent's predictions align with the desired outcome (goal prior). As illustrated in Figure 3a, the CARX-EFE agents consistently achieve better goal alignment over time, compared to the uncoupled ARX-EFE agents. Both agent sets exhibit initial peaks in the alignment error, reflecting difficulty in achieving goal alignment during the early stages of control. For the uncoupled agents, these peaks are notably larger and more frequent, reflecting greater initial instability and less effective goal adherence. Over time, the CARX-EFE agents maintain more stable and lower error values, suggesting a more robust alignment with the desired system state. Prediction error, measured by $-\log p(y_t \mid u_t)$, reflects the agent's ability to minimize surprise by accurately predicting system behavior based on control inputs. Figure 3b demonstrates that the CARX-EFE agents outperform the uncoupled ARX-EFE agents by consistently achieving lower surprise values. This suggests that CARX-EFE agents are more effective in learning the system dynamics and predicting the outcome of their actions, which in turn helps maintain better goal adherence. In contrast, the uncoupled agents, initially struggling with higher surprise values, demonstrate less accurate predictions over time.

Overall, CARX-EFE agents exhibit superior performance by improving stabilization, lower prediction variance, and more efficient control strategies compared to their uncoupled counterparts. These findings underscore the efficacy of the coupled approach in improving both the accuracy and stability of the control system, making CARX-EFE a more robust choice for managing complex dynamical systems.



(a) Goal alignment, measured by $-\log p(y_t|y_*)$, plotted over time. Coupled agents have better overall goal alignment, with less fluctuations compared to uncoupled agents.



(b) Prediction error (surprise), measured by the negative log-likelihood $-\log p(y_t \mid u_t)$, plotted over time. CARX-EFE agents achieve better performance by minimizing surprise more effectively.

Fig. 3: Comparison of model performance of a set of CARX-EFE agents versus a set of uncoupled ARX-EFE agents. Each subplot evaluates a specific aspect of performance: (a) goal alignment and (b) prediction error (surprise). Lower values indicate better performance. The top and bottom row in each subplot show the performance of agents controlling the first and second mass, respectively.

5 Discussion

Improved ability to stabilize and lower prediction variance demonstrated by CARX-EFE suggest a significant advantage in scenarios requiring reliable convergence, such as robotic control and adaptive systems in unpredictable environments. However, the current findings are based on a single simulation run, necessitating further validation. Conducting Monte Carlo experiments would confirm the robustness of CARX-EFE's advantages across varied conditions. Future work should also evaluate the CARX-EFE agents on nonlinear and underactuated systems, like a double pendulum or acrobot, to assess their ability to generalize. Additionally, benchmarking against other control methods could provide insights into the relative strength of CARX-EFE agents. The current implementation of CARX-EFE agents relies on a one-step ahead prediction, making their performance sensitive to the system update step size (Δt). Addressing this limitation by extending the prediction capability could reduce the dependence on these parameters, and possibly improve the efficiency of the coupled approach.

6 Conclusion

We investigated the control of a multi-joint mechanical system by coupling multiple autoregressive active inference agents that minimize expected free energy. We evaluate the effect of sharing data buffers (i.e., memories) in the autoregressive models of the agents. Our experiments demonstrate that coupling significantly improves the agent's ability to achieve both better goal alignment and lower prediction error. CARX-EFE agents consistently outperformed their uncoupled counterparts, showing lower prediction uncertainty with higher prediction accuracy (lower surprise), and greater long-term stability around the goal prior. It is important to note that the agent is limited to one-step ahead predictions. Future research should focus on extending the horizon of the agents, and improving the optimization procedure in MAP estimation.

Acknowledgments. The authors gratefully acknowledge support by the Eindhoven Artificial Intelligence Systems Institute and the Ministry of Education, Culture and Science of the Government of the Netherlands.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Baioumy, M., Duckworth, P., Lacerda, B., Hawes, N.: Active inference for integrated state-estimation, control, and learning. In: IEEE International Conference on Robotics and Automation. pp. 4665–4671 (2021)
2. Baltieri, M., Buckley, C.L.: Pid control as a process of active inference with linear generative models. Entropy **21**(3), 257 (2019)

3. Bucak, I.O., Zohdy, M.A.: Reinforcement learning control of nonlinear multi-link system. Engineering Applications of Artificial Intelligence **14**(5), 563–575 (2001)
4. Da Costa, L., Parr, T., Sajid, N., Veselic, S., Neacsu, V., Friston, K.: Active inference on discrete state-spaces: A synthesis. Journal of Mathematical Psychology **99**, 102447 (2020)
5. Friston, K., FitzGerald, T., Rigoli, F., Schwartenbeck, P., Pezzulo, G., et al.: Active inference and learning. Neuroscience & Biobehavioral Reviews **68**, 862–879 (2016)
6. Friston, K., Mattout, J., Trujillo-Barreto, N., Ashburner, J., Penny, W.: Variational free energy and the Laplace approximation. Neuroimage **34**(1), 220–234 (2007)
7. Friston, K., Rigoli, F., Ognibene, D., Mathys, C., Fitzgerald, T., Pezzulo, G.: Active inference and epistemic value. Cognitive neuroscience **6**(4), 187–214 (2015)
8. Friston, K.J., Daunizeau, J., Kilner, J., Kiebel, S.J.: Action and behavior: a free-energy formulation. Biological Cybernetics **102**(3), 227–260 (2010)
9. Gill, P.E., Murray, W., Wright, M.H.: Practical optimization. SIAM (2019)
10. van der Himst, O., Lanillos, P.: Deep active inference for partially observable MDPs. In: International Workshop on Active Inference. pp. 61–71 (2020)
11. Huebotter, J., Thill, S., Gerven, M.v., Lanillos, P.: Learning policies for continuous control via transition models. In: International Workshop on Active Inference. pp. 162–178. Springer (2023)
12. Imohiosen, A., Watson, J., Peters, J.: Active inference or control as inference? A unifying view. In: International Workshop on Active Inference. pp. 12–19. Springer (2020)
13. Kouw, W.M.: Information-seeking polynomial NARX model-predictive control through expected free energy minimization. IEEE Control Systems Letters (2023)
14. Krichmar, J.L.: Neurorobotics—a thriving community and a promising pathway toward intelligent cognitive robots. Frontiers in Neurorobotics **12**, 42 (2018)
15. van de Laar, T., Koudahl, M., van Erp, B., de Vries, B.: Active inference and epistemic value in graphical models. Frontiers in Robotics and AI **9**, 794464 (2022)
16. Lanillos, P., Meo, C., Pezzato, C., Meera, A.A., Baioumy, M., Ohata, W., Tschantz, A., Millidge, B., Wisse, M., Buckley, C.L., et al.: Active inference in robotics and artificial agents: Survey and challenges. arXiv:2112.01871 (2021)
17. Liagkou, V., Stylios, C., Pappa, L., Petunin, A.: Challenges and opportunities in industry 4.0 for mechatronics, artificial intelligence and cybernetics. Electronics **10**(16), 2001 (2021)
18. Lopes, M.T., Castello, D.A., Matt, C.F.T.: A bayesian inference approach to estimate elastic and damping parameters of a structure subjected to vibration tests. In: Proceedings of Inverse Problems, Design and Optimization Symposium (2010)
19. MacKay, D.J.: Information theory, inference and learning algorithms. Cambridge University Press (2003)
20. Massioni, P., Verhaegen, M.: Distributed control for identical dynamically coupled systems: A decomposition approach. IEEE Transactions on Automatic Control **54**(1), 124–135 (2009)
21. Parr, T., Friston, K., Zeidman, P.: Active data selection and information seeking. Algorithms **17**(3), 118 (2024)
22. Parr, T., Pezzulo, G., Friston, K.J.: Active inference: the free energy principle in mind, brain, and behavior. MIT Press (2022)
23. Pio-Lopez, L., Nizard, A., Friston, K., Pezzulo, G.: Active inference and robot control: a case study. Journal of The Royal Society Interface **13**(122), 20160616 (2016)
24. Press, W.H.: Numerical recipes: The art of scientific computing. Cambridge University Press (2007)

25. Särkkä, S.: Bayesian filtering and smoothing, vol. 3. Cambridge University Press (2013)
26. Schwartenbeck, P., Passecker, J., Hauser, T.U., FitzGerald, T.H., Kronbichler, M., Friston, K.J.: Computational mechanisms of curiosity and goal-directed exploration. eLife **8**, e41703 (2019)
27. Soch, J., Faulkenberry, T.J., Petrykowski, K., Allefeld, C.: The book of statistical proofs (2024). https://doi.org/10.5281/zenodo.4305949
28. Ueltzhöffer, K.: Deep active inference. Biological Cybernetics **112**(6), 547–573 (2018)
29. Williams, C.K., Rasmussen, C.E.: Gaussian Processes for Machine Learning. MIT Press (2006)

Modelling Agency Perception in a Multi-Agent Context in Depression Using Active Inference

Riddhi J. Pitliya[1,2], Dimitrije Marković[3], Federica Folesani[4], Martino Belvederi Murri[4], Santiago Castiello de Obeso[5], and Robin A. Murphy[2]

[1] VERSES Research Lab, Los Angeles, California, 90016, USA
[2] Department of Experimental Psychology, University of Oxford, Oxford, UK
[3] Chair of Cognitive Computational Neuroscience, Technische Universität Dresden, Dresden, Germany
[4] Institute of Psychiatry, Department of Neuroscience and Rehabilitation, University of Ferrara, Ferrara, Italy
[5] Department of Psychiatry, Yale University, New Haven, USA

Abstract. A reduced sense of agency is a primary symptom of depression. This study investigates how agency is learned and perceived in a multi-agent environment, comparing depressed and non-depressed individuals.

Participants explored their control over an on-screen outcome via button presses across multiple trials, while observing a simulated agent's button presses. They rated each agent's control after each block of trials. Experimental conditions varied which agent had control and the type/direction of control (positive/excitatory, negative/inhibitory, or none). We applied an active inference model to the behavioural data to understand action and perceptual processes in forming agentic beliefs.

Results showed all participants identified the controlling agent but incorrectly attributed control to the non-controlling agent in the opposite direction. Depressed participants consistently perceived themselves and others as having less agency, while viewing the other agent as having positive control across all conditions. The model suggested stronger prior beliefs about agents having opposing directions of control. Depressed participants, however, perceived their control to match that of external agents, while non-depressed participants perceived their control to be independent of the external agent.

Despite perceiving reduced agency, depressed participants increased their agentic action over time, suggesting heightened but potentially aimless environmental sampling. The model reflected this as a higher prior for acting and a lower preference for observing outcomes.

This study provides novel insights into how individuals with and without depression perceive agency differently within the same environment through differing biases in perceptual and action processes. Depressed individuals may not learn about other agents in the same way as non-depressed individuals. These findings could inform future therapeutic strategies and deepen our understanding of depression.

Keywords: sense of agency · depression · active inference · multi-agent · computational psychopathology

1 Introduction

Understanding agency - the extent to which individuals believe they have control over events in their environment - is crucial for effective decision-making and adaptive, goal-directed behaviour [1]. This sense of control shapes how agents interact with their world, influencing their choices, motivation, and overall well-being [2], [3]. Various models have been proposed to explain how individuals learn about their agency, encompassing the processes of perception, learning, and action.

Associative learning theories suggest that individuals perceive agency based on the strength of the connection, or *associative weight*, between their mental representations of actions and the resulting outcomes [4], [5]. In this framework, learning occurs when a prediction error prompts the updating of associative weights according to a learning rule. However, these models treat the learner as a passive recipient of information, with actions being conducted merely to gain rewards after the environment's associative structure has been learned.

Bayesian models may offer a more comprehensive framework for understanding agency and learning compared to statistical and associative approaches [6]. While associative models view actions as primarily reward-driven, Bayesian frameworks recognise that individuals often act in environments where the causal structure is unknown or uncertain. This perspective acknowledges that agents engage in information-seeking behaviour, conducting actions not just to achieve immediate goals, but also to generate data from which they can learn and refine their understanding of the environment [7], [8].

This dual purpose of actions - maximising rewards (exploitation) and maximising expected information gain (exploration) - is a key feature of Bayesian models that sets them apart from simpler approaches [9]. To facilitate exploration, Bayesian models represent beliefs as probability distributions, inherently capturing information uncertainty in a way that fixed associative weights cannot [10]. This nuanced representation of uncertainty enables Bayesian models to account for phenomena that challenge associative theories, such as rapid belief updates in volatile environments and the integration of prior knowledge with new evidence [11].

However, Bayesian models have limitations. They often optimise exploration and exploitation through separate functions, potentially necessitating ad-hoc parameters for specific tasks and reducing model flexibility [10]. Active inference, an extension of the Bayesian approach, addresses this by optimising a single metric: free energy [12]. This unification inherently balances exploration and exploitation [13], offering a potentially universal model for understanding behaviours, including agency formation. In active inference, individuals maintain generative models that guide actions, perception, and learning about the world. Unlike Bayesian agents, active inference agents possess a more comprehensive representation of the causal structure and dynamics of the environment, as their generative models incorporate beliefs about state transitions over time, as well as preferences for certain outcomes. This approach views perception and action as being more interdependent; actions are selected not only to achieve desired

outcomes but also to refine perception by reducing uncertainty and improving the agent's model of the world.

The current study investigated how perceived agency develops in the presence of other agents and why individuals may perceive agency differently within the same environment. This multi-agent approach more closely mirrors real-world scenarios, where multiple potential causes may compete to explain observed outcomes. We collected behavioural data using a complex version of our previous task [6], involving two agents (one participant and one simulated agent) and a single outcome. Our sample included individuals both with and without clinical depression, allowing us to examine variations in perceived agency.

Agency is transdiagnostic of mental health [14]. We chose to examine depression here because it is associated with a diminished perception of control [15], as evidenced by psychometric scales [16], [17] and empirical studies [18]. Several key propositions have been suggested to explain why depressed individuals experience reduced agency. Cognitive biases may lead depressed individuals to believe they have little or no control over their environment, even when they do [19]–[21]. Additionally or alternatively, depressed individuals may attribute greater control to other agents or external factors, further diminishing their sense of agency [18], [22], [23]. Finally the reduced engagement with the environment found in depressive individuals [24]–[26] may result in fewer opportunities to confirm control through clear action-outcome pairings [27], reinforcing the perception of diminished agency. These propositions are not necessarily mutually exclusive and may interact in complex ways to shape the reduced sense of agency observed in depression.

In this paper, we first describe the behavioural task and participant data. Next, we propose an active inference model of the task. Finally, we present the results from model fitting to compare the model's predictions with actual human behaviour and examine how the parameters vary among individuals with and without clinical depression, exploring how individuals may perceive agency differently in the same environment. This research has potential implications for understanding the cognitive mechanisms underlying depression and could inform the development of targeted interventions to enhance perceived agency in clinical populations.

2 Behavioural Data

Information on participants is in Appendix A.

2.1 Task Design

We employed a modified free-operant task (Figure 1) where participants freely produced actions, observed a simulated agent's actions, and integrated this information with the presence or absence of outcomes. Participants rated their perceived level of control over the outcome for both themselves and the simulated agent on a scale from -10 to 10. The outcome was a shape, either empty

(outline only) or filled (solid black). Participants were instructed to produce the filled shape as much as possible.

Each block consisted of 24 trials, lasting 1-3 seconds each. The simulated agent acted randomly 50% of the time, unknown to participants. After each block, participants rated perceived control for both agents. We recorded participants' actions to evaluate environmental sampling and alignment with inferred control. The task was delivered through the Gorilla platform [28].

The independent variables were: (i) which agent had control (the real or simulated participant) and (ii) the programmed contingency between the actions and outcomes (positive, zero, or negative). Positive contingency referred to their action causing the outcome, negative contingency referred to their action preventing the outcome, and zero meant their action had no apparent effect on the outcome. This resulted in five conditions: with the control condition being Zero, where no one has control, and the four experimental conditions being Self Positive (SP), Self Negative (SN), Other Positive (OP), and Other Negative (ON). When one agent had control, the other had zero control. This setup was designed to elicit any interaction effects between the two agents - for example, do people tend to believe that when the self has positive agency, the other has negative agency when in fact the other has none? All contingencies were deterministic, i.e., when an agent had control, they had 100% control.

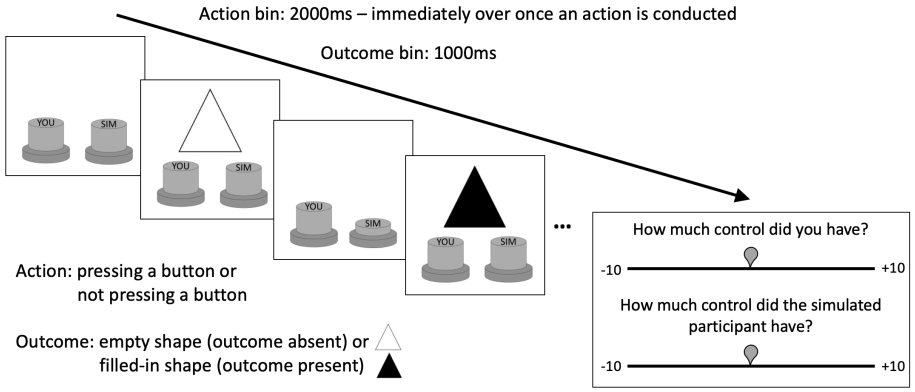


Fig. 1. An illustration of the two-agent agency learning task completed by participants.

2.2 Results

Results Regarding Ratings Figure 2 visualises the average agency ratings for clinically depressed and non-depressed participants under each experimental condition. Significant differences in ratings were found as a function of experimental condition, depression status, and self vs other ratings ($\chi^2(19) = 1185.3$,

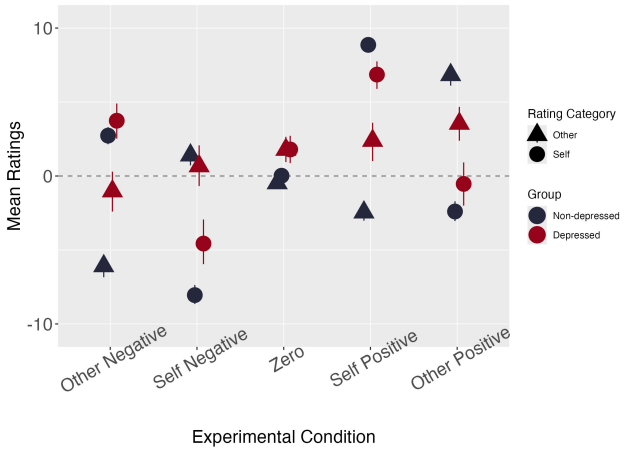


Fig. 2. The average ratings of agency attributed to the Self (represented by circles) and Other (the simulated agent; represented by triangles) across the experimental conditions. Ratings were provided by participants classified into two groups: clinically depressed (depicted in red) and non-depressed (depicted in black). Each point represents the average rating for a given condition, with error bars indicating the 95% confidence intervals

$p<0.001$). Follow-up pairwise comparisons reveal the key trends, which are described below. Details and results of the statistical analyses are provided in Appendix B.

In the Zero condition, only the non-depressed group correctly perceived that neither the self nor the other had any agency. As shown in Figure 2, the depressed group perceived more positive control for both the self and other.

Four main trends were identified in the experimental conditions. First, all participants learned which agent had control but incorrectly attributed control in the opposite direction to the non-controlling agent (e.g., if the participant was perceived as having positive control, the other agent was perceived as having negative control). Second, agency ratings for the self were consistently higher than for the other agent when the respective agent had control. Third, depressed participants generally perceived reduced overall agency for both the self and the other agent, as shown by the red points being nearer to the grey dotted line (representing value of zero) compared to the black points in Figure 2.

Finally, within this compressed range of perceived agency, depressed participants exhibited a bias toward perceiving more positive control, particularly for the other agent, across all experimental conditions. This effect is depicted in Figure 2 by the red triangle points all being close to or above zero, despite being closer to zero overall than the non-depressed participants' ratings. Notably, in the Other Negative condition, non-depressed participants correctly rated the other agent as having negative control, whereas depressed participants rated

them close to zero. Moreover, in the Self Positive condition, non-depressed participants rated the other agent as having negative control, following the first trend described, whilst depressed participants rated the other agent as also having positive control.

On average, collapsing across all conditions, depressed participants rated the agent at 1.470 (SE = 0.296) and themselves at 1.457 (SE = 0.345), compared to non-depressed participants, who gave lower, more balanced ratings: -0.159 (SE = 0.210) for the other and 0.232 (SE = 0.231) for the self. This suggests that non-depressed participants perceive their own agency as slightly positive and others' agency as slightly negative, but both averages are quite close to zero, suggesting more accurate perceptions. In contrast, depressed participants seem to overestimate positive agency for both themselves and others within their overall reduced range of perceived agency. This reflects a potential bias where depressed individuals perhaps struggle to process negative contingencies effectively, leading to a reduced but positively skewed perception of agency.

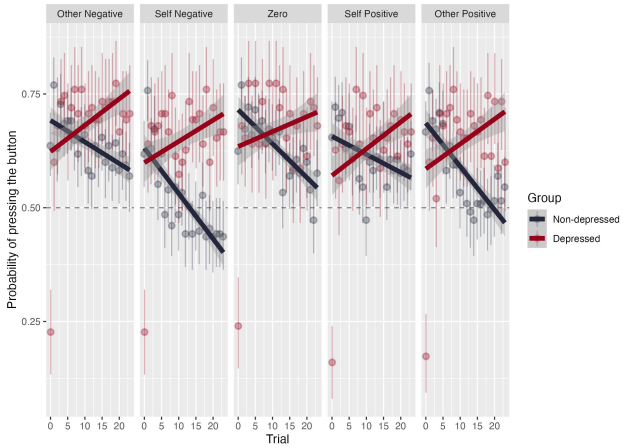


Fig. 3. The probability of pressing a button across a series of trials and experimental conditions for participants in clinically depressed (depicted in red) and non-depressed (depicted in black) groups. Points represent the average probability of pressing the button for each trial, with error bars indicating 95% confidence intervals. The lines represent the linear trends in button-pressing probability across trials for each group.

Results Regarding Actions Figure 3 illustrates the probability of producing an action (pressing the button), averaged across participants, under each experimental condition. A linear model analysis revealed an interaction effect between actions taken across time steps by depression group (β = 0.012, SE = 0.001,

t = 13.68, p<0.001). Across all experimental conditions, non-depressed participants tended to act more in earlier trials compared to depressed participants, and this effect is flipped in later trials. Across trials within each experimental condition, depressed participants increased the frequency of pressing the button (β = 0.005, SE = 0.001, t = 6.978, p<0.001), indicated by the upward trend of the red lines, suggesting they sampled the environment more as the trials progressed. Conversely, non-depressed participants decreased the frequency of pressing the button (β = -0.007, SE = 0.001, t = -14.220, p<0.001), as indicated by the downward trend of the black lines. Depressed participants tended not to press in the first trial across all experimental conditions as seen by the low red dots at trial 0.

3 Active Inference Model

In the first sub-section we describe the active inference model with its fixed parameters and the parameters varied during model fitting. Hypotheses regarding the parameter values are outlined given the behavioural results obtained. The active inference generative model used in this study, illustrated in Figure 4 in Appendix C, builds on the model presented in our previous paper [6]. Then, in the next two sub-sections, the model fitting procedure and results are presented, respectively.

3.1 The Active Inference Model

The active inference modelling was implemented using the JAX-based Python package, `pymdp`, which offers efficient and flexible tools for constructing such models [29]. For parameter inference we have used a PyBefit package which offers a convenient NumPyro [30], [31] wrapper for fitting computational behavioural models. The detailed code for both the model specification and the fitting process can be accessed here.

The hidden and controllable states in the model were classified into state factors, and observations were classified into observation modalities. At any given time, observations were generated from each modality, hidden states were inferred from each state factor, and an action (controllable state) was selected accordingly. The states (s), actions (u), and observations (o) were discrete random variables, making all model parameters categorical distributions.

At each time step (t), the focal agent had three observations: whether the simulated agent pressed the button or not ($o_t^{other_{action}}$); the action (μ) executed by the focal agent ($o_t^{self_{action}}$), which depended on the policy (π; single action policies) it selected; and the outcome ($o_t^{outcome}$), which could be absent or present depending on the experimental condition.

Based on these observations, the focal agent inferred four composite hidden states: Self Positive & Other Positive (SPOP), Self Positive & Other Negative (SPON), Self Negative & Other Positive (SNOP), Self Negative & Other Negative (SNON). These were derived from combinations of individual agency states (positive, negative, zero) for both self and other. These four states efficiently represented all possible agency scenarios and were used to compute

the model's equivalent of participants' ratings: self-ratings as (SPOP+SPON) – (SNOP+SNON) and other-ratings as (SPOP+SNOP) – (SPON+SNON).

The agent's generative model included parameters represented by tensors **A**, **B**, **C**, **D**, and **E**. The likelihood tensor **A** was initialised with probabilities p_1, 0.5, and p_2, such that $0.0 < p_1 < 0.5$ and $1.0 > p_2 > 0.5$, with p_1 and p_2 treated as free parameters to be fit to the data. These values reflected the agent's belief about the presence or absence of the outcome given different state combinations. For instance, at the start of a trial block, the agent might believe there was a 0.75 probability of observing the outcome when in the SPOP state with both the focal and other agent pressing the button. The likelihood tensor represented the agent's uncertain beliefs, in contrast to the deterministic nature of the experimental conditions unknown to participants. As beliefs updated over time, the precision of beliefs over states would increase, resulting in stronger state beliefs reflected in higher ratings. We hypothesised a lower parameter value for the depressed group. This lower value would result in a flatter probability distribution in the likelihood mapping, reflecting less certainty in the relationship between states and outcomes, and consequently leading to reduced ratings across all conditions.

The **D** tensor represented the agent's prior beliefs about states. Priors for self and other's actions were uniformly distributed, while priors for the s_t^{agency} state factor were determined through model fitting. We hypothesised stronger prior beliefs for SPON and SNOP states in both groups, reflecting a tendency to believe that when one agent has positive agency, the other has negative agency. Additionally, we predicted that depressed participants would show stronger prior beliefs for the SPOP state compared to non-depressed participants, suggesting a bias towards assuming both agents have positive control across all conditions.

The state transition tensor (**B**) represented beliefs about how hidden states and actions determined subsequent hidden states, $P(s_t|s_{t-1}, u_t)$. The experimental condition remained constant over a block of trials, and it was assumed that the agent knew this (i.e., identity matrix). The **B** tensor for self-action was set as a fully controllable state factor, and the one for other-action was set as a flat distribution to reflect the belief that the other agent could take any action. Policies (π; action sequences) were inferred according to the possible actions.

The **E** tensor represented the prior probability of selecting a policy, conceptualised as a habit. Policy selection was based on expected free energy (EFE) calculations, with the influence of habits modulated by gamma. Lower gamma values flattened the precision of policies under EFE calculations, giving more weight to **E** in policy selection. Both the habit value for producing actions and the gamma value were determined through model fitting. We hypothesised that, compared to the non-depressed group, participants in the depression group would have a stronger habit to press the button and a lower gamma value. This would reflect their tendency to increasingly press the button across trials and experimental conditions.

Furthermore, in this model, the agent had a preference to see the outcome present as participants were instructed to produce the filled shape (i.e., the out-

come) as often as possible. The degree of preference was determined by the model fitting process. As research has shown that depressed individuals exhibit diminished reward processing [32]–[34], it was hypothesised that participants with depression would have a lower preference to see the outcome present compared to non-depressed participants. This may also explain the slopes in Figure 3, starting off with producing actions to produce the outcome according to their preferences (low for depressed participants), and changing over trials as habit takes over (increasing for depressed participants).

3.2 Model Fitting Procedure

The model fitting process for both groups employed stochastic variational inference (SVI), a probabilistic framework that approximates posterior distributions of model parameters. This approach allowed us to understand parameter variations across groups and identify differences in cognitive processes. The free parameters included the prior over agency, habit to act (button pressing), gamma, preference for seeing the outcome present, probability of observing the outcome present in the likelihood tensor, and the standard deviation used to fit ratings according to beliefs.

As a parametric prior we used a hierarchical setup, with a Regularised Horseshoe prior [35] for group level uncertainty, Gaussian prior for group level mean, and a Gaussian prior conditioned on group level mean and uncertainty for individual subject-specific parameters. The hierarchical setup provides a balance between group-level trends and individual differences. This hierarchical prior results in estimates pulled toward group level mean, when individual differences are non-inferrable from the data, while at the same time allowing for substantial individual differences in parameter estimates when data provides sufficient evidence.

The fitting process combined this prior with a likelihood function measuring the alignment between model predictions and observed data. SVI was run for 5,000 iterations, adjusting model parameters to minimise the Evidence Lower Bound (ELBO). The loss function, which showed good convergence, was plotted across iterations to assess the fitting process; the plot can be viewed here. Post-fitting, we analysed the posterior samples to compare parameters between the depressed and non-depressed groups.

3.3 Results

The model fitting results, summarised in Table 1, revealed interesting trends in the parameter sets between clinically depressed (n=25) and non-depressed (n=55) agents. These trends align with our hypotheses and offer valuable insights, despite the unequal and relatively small sample sizes. The full analysis, including 95% credible intervals, Bayesian group differences, and effect sizes (Cohen's d), can be viewed here. These findings provide a foundation for future research into agency perception and learning in depression.

Parameter	Depressed Group	Non-Depressed Group
P(outcome_present) for likelihood tensor	0.547	0.565
Stdev for ratings regarding self	0.288	0.246
Stdev for ratings regarding other	0.305	0.286
Preference to observe outcome present	0.551	1.689
Habit to press the button	0.687	0.594
Prior for SPOP	0.280	0.159
Prior for SPON	0.288	0.368
Prior for SNOP	0.290	0.303
Prior for SNON	0.142	0.171

Table 1. Comparison of parameter values between Depressed and Non-Depressed groups.

All agents held prior beliefs over SPON and SNOP, reflecting a bias where agency in one direction implies the opposite direction for the other agent. Depressed agents exhibited a stronger prior belief about the SPOP state, while non-depressed agents had a stronger prior belief about the SPON state. The priors also indicated that depressed agents believed both self and others had more positive control (self: (SPOP + SPON) – (SNOP + SNON) = 0.136; others: (SPOP + SNOP) – (SPON + SNON) = 0.140), while non-depressed agents were less biased, with a slight tendency for self having more positive agency and others more negative agency (self: 0.053, others: -0.077).

Moreover, the likelihood tensor had slightly lower values for the probability of outcome being present (i.e., flatter distribution) and higher standard deviations of the fit around the ratings in the depressed group, indicating less precise beliefs and reflecting the reduced perceived agency across all conditions.

Additionally, non-depressed agents preferred to observe the outcome present more strongly than depressed participants, and had a lower habit of pressing the button. The depressed group also had a lower gamma value, suggesting that they were more influenced by their habits.

4 Discussion and Concluding Remarks

Our study investigated how clinically depressed and non-depressed individuals learn and perceive agency in a multi-agent environment. Using a novel behavioural task and an active inference model, we gained insights into the complex interactions between agents in relation to agency perception. Our findings suggest that the presence of another agent may alter how individuals interpret their own agency and that of others, contributing to the understanding of agency in social contexts and extending beyond the single-agent scenarios that have been the focus of much previous research.

The well-established link between depression and reduced agency perception [15], [18] was reflected in our study. Depressed participants reported reduced

agency for both self and other agents across all conditions. The active inference model captured this through lower probability values of outcome presence in the likelihood tensor, resulting in a flatter distribution. This implies that depressed participants may be less certain about their agency given the same observations, aligning with findings of reduced certainty or confidence in other cognitive behavioural tasks [36], [37].

Moreover, a key finding is the differential attribution of control to self and other agents between depressed and non-depressed individuals. Depressed participants tended to perceive others as having positive control, aligning with research linking depressive symptoms to an external locus of control [18], [22], [23]. Interestingly, they also perceived themselves as having positive control, matching that of external agents. In contrast, non-depressed participants perceived their control as independent of the external agent, and in more opposing directions of control. This pattern may be attributed to several interrelated factors in depressed individuals: a blurred self-other distinction, which has been associated with increased mirroring behaviour; a reduced sense of agency; and potentially blunted cognitive processing of complex social scenarios involving theory of mind [38]. These factors could collectively contribute to more homogeneous attributions of control.

Our data revealed potential differences in observation preferences and environmental engagement. Non-depressed participants showed a higher preference for observing the outcome, aligning with the link between depression and blunted sensitivity to goals and rewards [32]–[34]. Depressed participants exhibited a trend towards button-pressing, suggesting heightened but potentially aimless environmental sampling. This relative homogeneity of action patterns across conditions (Figure 3) may indicate non-adaptive environmental responses, consistent with findings of disengagement and aimlessness in depression [39], [40]. The increased button-pressing could also be interpreted as a compensatory mechanism, where depressed individuals are trying to gain more information about their environment due to heightened uncertainty. In contrast, non-depressed participants tended to press the button less frequently, possibly due to quicker, more confident learning about agency.

The increased button-pressing observed in depressed participants might also be explained by differences in perceived control. While depressed participants viewed their control as matching that of external agents, non-depressed participants perceived their control as independent. This perception of equivalent control with external agents might have driven depressed participants to engage in more frequent actions, perhaps in an attempt to exert or confirm this perceived level of influence over their environment.

The tendency for clinically depressed participants to not produce an action at the first timestep is intriguing. While this could be due to delayed processing or attention [41], it might also reflect a strategic approach of observing before acting. This hesitation could be adaptive in uncertain environments, suggesting that depressed individuals might be more cautious in novel situations. Further

research could manipulate task instructions to explore whether this hesitation is strategic or a byproduct of cognitive slowing.

These findings underscore the importance of considering agency perception within a social context, revealing complexities in agency attribution and behavioural strategies not apparent in single-agent paradigms. This provides a more ecologically valid model of how individuals, particularly those with depression, navigate agency in real-world social environments.

Nonetheless, there are limitations of our study to consider. The simplified agency representation, with only one causal structure (two potential causes and one outcome), limits generalisability. Real-world environments often involve multiple causes and outcomes, which were not explored in this study. The absence of an emotional context further limits ecological validity. Additionally, while we identified group-level trends, our study does not account for the heterogeneity within depressed and non-depressed groups. Moreover, the relatively small and unequal sample sizes (25 depressed and 55 non-depressed participants) likely contributed to the lack of definitive statistical evidence for group differences. Furthermore, our study's cross-sectional nature limits our ability to infer causal relationships between depression and agency perception.

Future research should address these limitations by incorporating more complex causal structures, multiple outcomes, and emotionally relevant contexts. Studies should also explore individual differences within groups, acknowledging the diversity of experiences within depression. Larger and more balanced sample sizes would increase statistical power and the ability to detect subtle group differences. Longitudinal designs could help elucidate the causal relationships between depression and changes in agency perception over time, perhaps by including learning of the likelihood tensor (via Dirichlet counts), under different contexts.

Our novel behavioural task and active inference modelling highlight the intricate relationship between perceptual and action processes that point to the interplay between agency inference and depression in multi-agent contexts. While our results are not definitive, they provide a foundation for future research. By elucidating the mechanisms underlying agency inference in depression within social contexts, we can better inform therapeutic strategies. Cognitive-behavioural therapies could focus on helping depressed individuals recognise and challenge their beliefs about agency in social contexts, focusing on differentiating and highlighting the independence of the self and other. Mindfulness-based interventions might increase awareness of one's actions and their consequences, potentially addressing the tendency towards aimless environmental sampling and independence of self. Our research approach can be extended to assess agential learning in multi-agent contexts across various conditions associated with altered senses of agency [42]–[45], potentially leading to more tailored interventions across a range of mental health conditions.

References

[1] P. F. Verschure, C. M. Pennartz, and G. Pezzulo, "The why, what, where, when and how of goal-directed choice: Neuronal and computational principles," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 369, no. 1655, p. 20 130 483, 2014.

[2] A. S. Kayser, J. M. Mitchell, D. Weinstein, and M. J. Frank, "Dopamine, locus of control, and the exploration-exploitation tradeoff," *Neuropsychopharmacology*, vol. 40, no. 2, pp. 454–462, 2015.

[3] T. Penton, X. Wang, M.-P. Coll, C. Catmur, and G. Bird, "The influence of action–outcome contingency on motivation from control," *Experimental brain research*, vol. 236, pp. 3239–3249, 2018.

[4] R. E. Cramer, R. F. Weiss, R. William, S. Reid, L. Nieri, and B. Manning-Ryan, "Human agency and associative learning: Pavlovian principles govern social process in causal relationship detection," *The Quarterly Journal of Experimental Psychology: Section B*, vol. 55, no. 3, pp. 241–266, 2002, ISSN: 0272-4995.

[5] J. W. Moore, A. Dickinson, and P. C. Fletcher, "Sense of agency, associative learning, and schizotypy," *Consciousness and Cognition*, vol. 20, no. 3, pp. 792–800, 2011, ISSN: 1053-8100. DOI: `https://doi.org/10.1016/j.concog.2011.01.002`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S1053810011000031`.

[6] R. J. Pitliya and R. A. Murphy, "A model of agential learning using active inference," in *International Workshop on Active Inference*, Springer, 2023, pp. 106–120.

[7] A. Y.-C. Chang, H. Oi, T. Maeda, and W. Wen, "The sense of agency from active causal inference," *bioRxiv*, pp. 2024–01, 2024.

[8] J. Hohwy, "The self-evidencing brain," *Noûs*, vol. 50, no. 2, pp. 259–285, 2016, ISSN: 0029-4624.

[9] J. D. Cohen, S. M. McClure, and A. J. Yu, "Should i stay or should i go? how the human brain manages the trade-off between exploitation and exploration," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 362, no. 1481, pp. 933–942, 2007.

[10] J. K. Kruschke, "Bayesian approaches to associative learning: From passive to active learning," *Learning & behavior*, vol. 36, no. 3, pp. 210–226, 2008.

[11] T. E. Behrens, M. W. Woolrich, M. E. Walton, and M. F. Rushworth, "Learning the value of information in an uncertain world," *Nature neuroscience*, vol. 10, no. 9, pp. 1214–1221, 2007.

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

[13] D. Marković, H. Stojić, S. Schwöbel, and S. J. Kiebel, "An empirical evaluation of active inference in multi-armed bandits," *Neural Networks*, vol. 144, pp. 229–246, 2021.

[14] R. Murphy, N. Byrom, and R. M. Msetfi, "The problem with explaining symptoms: The origin of biases in causal processing," *European Journal for Person Centered Healthcare*, vol. 5, no. 3, 2017.

[15] M. E. Seligman, *Depression and learned helplessness.* John Wiley & Sons, 1974.
[16] D. M. Fresco, L. B. Alloy, and N. Reilly–Harrington, "Association of attributional style for negative and positive events and the occurrence of life events with depression and anxiety," *Journal of social and clinical psychology*, vol. 25, no. 10, pp. 1140–1160, 2006, ISSN: 0736-7236.
[17] A. Tapal, E. Oren, R. Dar, and B. Eitam, "The sense of agency scale: A measure of consciously perceived control over one's mind, body, and the immediate environment," *Frontiers in psychology*, vol. 8, p. 1552, 2017, ISSN: 1664-1078.
[18] S. Castiello, S. Senan, R. M. Msetfi, and R. A. Murphy, "Traits for depression related to agentic and external control," *Learning and Motivation*, vol. 72, p. 101 684, 2020.
[19] L. B. Alloy and L. Y. Abramson, "Judgment of contingency in depressed and nondepressed students: Sadder but wiser?" *Journal of experimental psychology: General*, vol. 108, no. 4, p. 441, 1979.
[20] R. Ackermann and R. J. DeRubeis, "Is depressive realism real?" *Clinical Psychology Review*, vol. 11, no. 5, pp. 565–584, 1991.
[21] S. F. Maier and M. E. Seligman, "Learned helplessness: Theory and evidence.," *Journal of experimental psychology: general*, vol. 105, no. 1, p. 3, 1976.
[22] P. A. Aiken and D. H. Baucom, "Locus of control and depression: That confounded relationship," *Journal of Personality Assessment*, vol. 46, no. 4, pp. 391–395, 1982.
[23] V. A. Benassi, P. D. Sweeney, and C. L. Dufour, "Is there a relation between locus of control orientation and depression?" *Journal of abnormal psychology*, vol. 97, no. 3, p. 357, 1988.
[24] F. Blanco, H. Matute, and M. A. Vadillo, "Mediating role of activity level in the depressive realism effect," 2012.
[25] F. Blanco, H. Matute, and M. A. Vadillo, "Depressive realism: Wiser or quieter?" *The Psychological Record*, vol. 59, no. 4, pp. 551–562, 2009.
[26] P. M. Lewinsohn, J. M. Sullivan, and S. J. Grosscup, "Changing reinforcing events: An approach to the treatment of depression.," *Psychotherapy: Theory, Research & Practice*, vol. 17, no. 3, p. 322, 1980.
[27] H. Matute, "Illusion of control: Detecting response-outcome independence in analytic but not in naturalistic conditions," *Psychological Science*, vol. 7, no. 5, pp. 289–293, 1996.
[28] A. L. Anwyl-Irvine, J. Massonnié, A. Flitton, N. Kirkham, and J. K. Evershed, "Gorilla in our midst: An online behavioral experiment builder," *Behavior Research Methods*, vol. 52, no. 1, pp. 388–407, 2020, ISSN: 1554-3528. DOI: 10.3758/s13428-019-01237-x. [Online]. Available: https://doi.org/10.3758/s13428-019-01237-x.
[29] C. Heins, B. Millidge, D. Demekas, *et al.*, "Pymdp: A python library for active inference in discrete state spaces," *arXiv preprint arXiv:2201.03904*, 2022.

[30] D. Phan, N. Pradhan, and M. Jankowiak, "Composable effects for flexible and accelerated probabilistic programming in numpyro," *arXiv preprint arXiv:1912.11554*, 2019.

[31] E. Bingham, J. P. Chen, M. Jankowiak, *et al.*, "Pyro: Deep universal probabilistic programming," *J. Mach. Learn. Res.*, vol. 20, 28:1–28:6, 2019. [Online]. Available: http://jmlr.org/papers/v20/18-403.html.

[32] R. J. Pitliya, B. D. Nelson, G. Hajcak, and J. Jin, "Drift-diffusion model reveals impaired reward-based perceptual decision-making processes associated with depression in late childhood and early adolescent girls," *Research on Child and Adolescent Psychopathology*, pp. 1–14, 2022, ISSN: 2730-7174.

[33] D. A. Pizzagalli, A. J. Holmes, D. G. Dillon, *et al.*, "Reduced caudate and nucleus accumbens response to rewards in unmedicated individuals with major depressive disorder," *American Journal of Psychiatry*, vol. 166, no. 6, pp. 702–710, 2009, ISSN: 0002-953X.

[34] E. Vrieze, D. A. Pizzagalli, K. Demyttenaere, *et al.*, "Reduced reward learning predicts outcome in major depressive disorder," *Biol Psychiatry*, vol. 73, no. 7, pp. 639–45, 2013, ISSN: 1873-2402 (Electronic) 0006-3223 (Linking). DOI: 10.1016/j.biopsych.2012.10.014. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/23228328.

[35] J. Piironen and A. Vehtari, "Sparsity information and regularization in the horseshoe and other shrinkage priors," 2017.

[36] T. Fu, W. Koutstaal, C. H. Fu, L. Poon, and A. J. Cleare, "Depression, confidence, and decision: Evidence against depressive realism," *Journal of Psychopathology and Behavioral Assessment*, vol. 27, pp. 243–252, 2005, ISSN: 0882-2689.

[37] J. E. Herskovic, M. L. Kietzman, and S. Sutton, "Visual flicker in depression: Response criteria, confidence ratings and response times," *Psychological Medicine*, vol. 16, no. 1, pp. 187–197, 1986, ISSN: 0033-2917. DOI: 10.1017/S0033291700002622. [Online]. Available: https://www.cambridge.org/core/product/3D233E799B555895183D1A44105930BE.

[38] C. M. Eddy, "The transdiagnostic relevance of self-other distinction to psychiatry spans emotional, cognitive and motor domains," *Frontiers in Psychiatry*, vol. 13, p. 797 952, 2022.

[39] S. Aoki, S. Doi, S. Horiuchi, *et al.*, "Mediating effect of environmental rewards on the relation between goal-directed behaviour and anhedonia," *Current Psychology*, vol. 40, no. 8, pp. 3651–3658, 2021, ISSN: 1936-4733. DOI: 10.1007/s12144-019-00312-y. [Online]. Available: https://doi.org/10.1007/s12144-019-00312-y.

[40] R. Levy, "Apathy: A pathology of goal-directed behaviour. a new concept of the clinic and pathophysiology of apathy," *Revue Neurologique*, vol. 168, no. 8, pp. 585–597, 2012, ISSN: 0035-3787. DOI: https://doi.org/10.1016/j.neurol.2012.05.003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0035378712008703.

[41] D. American Psychiatric Association, D. American Psychiatric Association, *et al.*, *Diagnostic and statistical manual of mental disorders: DSM-5*. American psychiatric association Washington, DC, 2013, vol. 5.
[42] J. Szalai, "The sense of agency in ocd," *Review of Philosophy and Psychology*, vol. 10, no. 2, pp. 363–380, 2019.
[43] A. Ciaunica, A. Seth, J. Limanowski, C. Hesp, and K. J. Friston, "I overthink—therefore i am not: An active inference account of altered sense of self and agency in depersonalisation disorder," *Consciousness and Cognition*, vol. 101, p. 103 320, 2022.
[44] C. D. Frith, "The self in action: Lessons from delusions of control," *Consciousness and cognition*, vol. 14, no. 4, pp. 752–770, 2005, ISSN: 1053-8100.
[45] C. D. Frith, S.-J. Blakemore, and D. M. Wolpert, "Explaining the symptoms of schizophrenia: Abnormalities in the awareness of action," *Brain Research Reviews*, vol. 31, no. 2-3, pp. 357–363, 2000, ISSN: 0165-0173.
[46] A. T. Beck, R. A. Steer, and G. K. Brown, "Beck depression inventory," 1996.
[47] M. von Glischinski, R. von Brachel, and G. Hirschfeld, "How depressed is "depressed"? a systematic review and diagnostic meta-analysis of optimal cut points for the beck depression inventory revised (bdi-ii)," *Quality of Life Research*, vol. 28, pp. 1111–1118, 2019.

5 Appendix A

A total of 87 participants, self-reporting as free from photosensitive epilepsy, were initially recruited. Seven participants were excluded due to uniform ratings across all conditions, indicating a lack of sensitivity to task manipulations. The final sample for data analysis comprised 80 participants (30 males, 47 females, 2 non-binary, 1 undisclosed; mean age = 32.6 years, SD = 14.8).

Participants were recruited from three sources and completed Beck's Depression Inventory (BDI-II, [46]); the table below summarises the demographic information from the three sources. The BDI-II is a widely used self-report measure of depression severity, with scores ranging from 0 to 63. To aid in interpreting the BDI-II scores, it is worth noting that a score of 13 or above is often used to indicate the presence of depressive symptoms [46], [47]. However, this threshold is not used as a diagnostic criterion in our study and is provided solely for reference.

Sample source	In-patient depressed individuals from a clinic in Italy	Not clinically depressed individuals from Italy	Not clinically depressed undergraduate students from a university in the UK
Total number of participants	25	24	31
Number of male participants	10	9	11
Number of female participants	15	14	18
Number of participants who identify with neither of the above two genders	0	1	1
Mean age	46.0	35.5	19.5
Standard deviation of age	12.2	12.6	1.5
Mean BDI-II score	29.7	5.0	10.0
Standard deviation of BDI-II score	11.5	5.3	11.2

6 Appendix B

Statistical analyses and results for the behavioural data.

Tests for normality, homogeneity of variances, and independence of observations were conducted to assess which analysis to conduct to assess if the group means were statistically different. The Shapiro-Wilk test indicated that the residuals significantly deviated from normality, W = 0.96601, $p<0.001$. Bartlett's test for homogeneity of variances indicated significant differences in variances across groups, Bartlett's K-squared = 365.12, df = 19, $p<0.001$. The independence of observations was guaranteed by the task design, but a residual plot was created to verify this. The residual plot showed no clear pattern or trend, suggesting that the assumption of independence was met. This was further supported by the Durbin-Watson test, which showed no significant autocorrelation in the residuals (D-W statistic = 1.95, $p = 0.262$).

Due to the violations of normality and homogeneity of variances, the Kruskal-Wallis test was used to compare group means. Results indicated significant differences in the rating distributions among the groups defined by the interaction of experimental condition, depressed vs non-depressed group, and the ratings of self vs other ($\chi^2(19) = 1185.3$, $p<0.001$). Follow-up pairwise comparisons using the Wilcoxon rank-sum test with Bonferroni correction were conducted to identify specific group differences.

In the Zero condition, the control experimental condition, only the non-depressed group perceived that neither the self nor the other had any agency. As shown in Figure 2, the depressed group perceived slightly more positive control for both the self and other, indicated by the higher red points compared to the black points. Pairwise comparisons partially supported this, revealing that the ratings provided by the depressed group significantly differed from the ratings provided by the non-depressed group regarding the self, but the not the other.

In experimental conditions other than Zero, four main trends were identified. First, both depressed and non-depressed participants correctly learned which agent had control but perceived the non-controlling agent as having control in the opposite direction. Recall that in the experimental conditions, when one agent had control, the other agent had zero control. Pairwise comparisons revealed that this trend is seen across both groups across all experimental conditions, except in the depressed group in the Self Positive and Other Negative condition. This exception may be explained by the bias of other agents having positive control (the last trend mentioned here).

Second, across all participants, agency ratings for the self were higher than for others when the respective agents had control. This is reflected in Figure 2, where, for example, the rating for the self (circles) in the Self Positive condition was higher than the rating for the other (triangles) in the Other Positive condition. This effect is partially reflected in the pairwise statistical analysis, where in the Positive conditions, the ratings for self were significantly higher in the depressed group but not in the non-depressed group. Conversely, in the Negative conditions, the ratings for the self were significantly more negative in the non-depressed group compared to the depressed group.

Third, the depressed group generally perceived reduced agency of the self and other. This is reflected in Figure 2 as, compared to the black points, the red points are generally closer to the grey dotted line, representing ratings of the value zero. Pairwise comparisons showed statistically significant differences between the ratings provided by the depressed and non-depressed groups regarding the agent that had control in that experimental condition. There were no statistically significant differences between the depressed and non-depressed groups' ratings regarding the agent that did not have control in that experimental condition, except for Self Positive.

Finally, within the reduced perceived agency ratings, depressed participants exhibited a bias toward perceiving other agents as having more positive control across all experimental conditions. This is depicted in Figure 2 by the red triangle points all being close to or above zero. Notably, in Other Negative, non-depressed participants correctly rated the other as having negative control, however depressed participants rated close to zero. Moreover, in Self Positive, non-depressed participants rated the other as having negative control, following the first trend described, however, depressed participants rated the other as also having positive control. Pairwise comparisons also support this finding, as depressed participants' ratings regarding the other agent did not significantly differ across experimental groups, except between Other Positive and Other Negative. This is contrasted by non-depressed participants' ratings regarding the other agent significantly differing across all experimental groups.

Furthermore, on average, collapsing across all conditions, depressed participants rated the other agent at 1.470 (SE = 0.296) and the self at 1.457 (SE = 0.345), compared to non-depressed participants, who gave lower, more balanced ratings: -0.159 (SE = 0.210) for the other and 0.232 (SE = 0.231) for the self. This suggests that non-depressed participants perceive their own agency as slightly positive and others' agency as slightly negative, but both averages are quite close to zero, indicating more accurate perceptions. In contrast, depressed participants seem to overestimate agency for both themselves and others, reflecting a potential bias where they perhaps struggle to process negative contingencies effectively.

7 Appendix C

Supplementary Figures and Tables

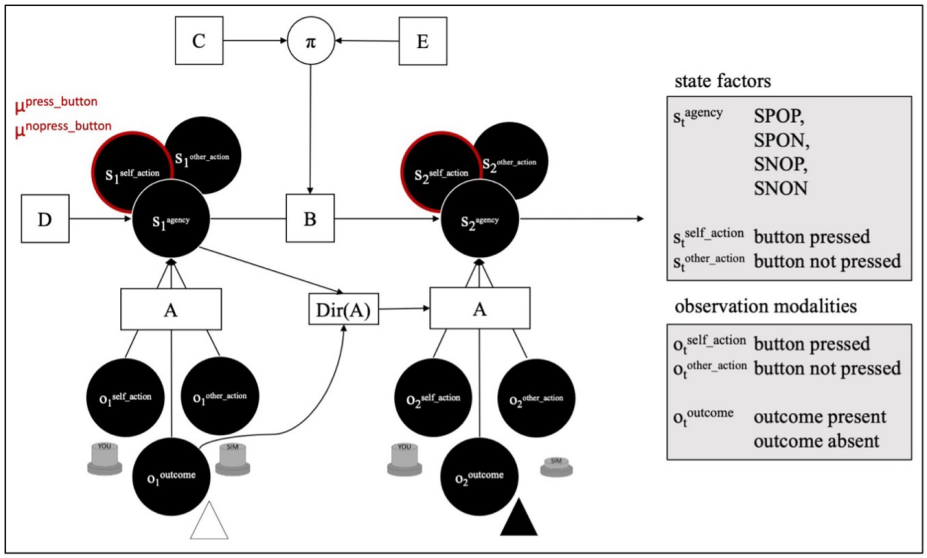


Fig. 4. A graphical representation of the active inference-based generative model of the two-agent agential learning task. The states inferred in the state factor of agency (s_t^{agency}) are Self Positive Other Positive (SPOP), Self Positive Other Negative (SPON), Self Negative Other Positive (SNOP), and Self Negative Other Negative (SNON). The red outlined state factor indicates that it is controllable by the focal agent, with the red text beside it showing the two actions the agent can take. The model's variables are depicted as circles, while the parameters are shown as squares and rectangles. Arrows indicate the direction of influence. For a detailed description of the variables and parameters, please refer to the main text.

Learning and embodied decisions in active inference

Matteo Priorelli, Ivilin Peev Stoianov, Giovanni Pezzulo[1]

Institute of Cognitive Sciences and Technologies, National Research Council of Italy

Abstract. Biological organisms constantly face the necessity to act timely in dynamic environments and balance choice accuracy against the risk of missing valid opportunities. As formalized by embodied decision models, this might require brain architectures wherein decision-making and motor control interact reciprocally, in stark contrast to traditional models that view them as serial processes. Previous studies have assessed that embodied decision dynamics emerge naturally under active inference – a computational paradigm that considers action and perception as subject to the same imperative of free energy minimization. In particular, agents can infer their targets by using their own movements (and not only external sensations) as evidence, i.e., via *self-evidencing*. Such models have shown that under appropriate conditions, action-generated feedback can stabilize and improve decision processes. However, how adaptation of internal models to environmental contingencies influences embodied decisions is yet to be addressed. To shed light on this challenge, in this study we systematically investigate the learning dynamics of an embodied model of decision-making during a *two-alternative forced choice* task, using a hybrid (discrete and continuous) active inference framework. Our results show that active inference agents can adapt to embodied contexts by learning various statistical regularities of the task – namely, prior preferences for the correct target, cue validity, and response strategies that prioritize faster or slower (but more accurate) decisions. Crucially, these results illustrate the efficacy of learning discrete preferences and strategies using sensorimotor feedback from continuous dynamics.

Keywords: active inference · hybrid models · embodied decisions · motor inference · motor learning

1 Introduction

The study of value-based decision-making in psychology and neuroscience has often focused on static settings, in which participants are asked to choose between a fixed number of choice alternatives (usually two) on the basis of known attributes, such as probabilities, utility, temporal delay, or their combinations. These situations are generally characterized in terms of a family of serial (*decide-then-act*) models, in which a decision is firstly made upon the accumulation of sensory evidence to a threshold and then reported by acting, e.g., by pressing a response button. Serial models such as the drift-diffusion model, the leaky, competing

accumulator model, or the race model have been highly successful in explaining a wide variety of human behavioral and neural data, during value-based and perceptual decisions [36,41].

Living organisms, however, often face *embodied decisions* that differ significantly from the value-based decisions widely studied in the laboratory [7,20,44]. Consider a lion chasing gazelles in the savanna or a driver surpassing other drivers while avoiding collisions. The embodied character of the decisions is mostly evident from the fact that agents make choices not just about potential outcomes but also about potential action plans to achieve them – or between competing affordances for movement, i.e., an affordance competition process [8,5,27]. Environments can also change rapidly and require updating of action plans on the fly. Thus, agents must implement these plans promptly to avoid losing valued opportunities. Differently from static decisions studied in the laboratory, embodied decisions can regard an open-ended number of choice alternatives and features that are not necessarily defined (or known) a priori and that might be continuous and change over time.

Various empirical studies addressed embodied decisions, both in minimally embodied setups in which classical decision tasks are augmented to require simple action dynamics, and in more sophisticated setups that mimic more closely the competition between movement affordances faced by animals in their lives [4,14,38,42]. These studies reveal that the serial (*decide-then-act*) view is insufficient to fully account for embodied decisions, for two main reasons. First, action and decision dynamics can unfold in parallel, in such a way that movement dynamics provide a rich readout of the ongoing decision [13,39]. Second, and perhaps more importantly, action dynamics can influence choice, e.g., in terms of motor costs associated with the alternatives [24,2,10,22].

Some of these findings are well accounted for by a novel class of embodied decision models that go beyond serial assumptions. For example, the *affordance competition* model assumes that the brain can specify, evaluate, prepare and sometimes even execute potential action plans in parallel [8,5,27]; in turn, this might require distributing the burden of decision-making across several circuits and networks, therefore implementing choice as a distributed consensus rather than as a centralized process [6]. Other embodied decision models also incorporate feedback from action to decision-making, simultaneously optimizing decision and action processes [4,23]. A recent computational study [35] showed that embodied decision dynamics emerge naturally under active inference, a computational paradigm that considers action and perception as subject to the same imperative of free energy minimization [26]. Key to this model is the reciprocal loop between *motor planning* – during which beliefs about the target to be reached contribute to selecting an appropriate motor plan – and *motor inference* – during which target-directed plans and movements are used as evidence to update beliefs about the target to be reached, in parallel to sensory evidence. Under appropriate conditions, this continuous interaction can stabilize and improve decisions.

However, that study was based on a fixed generative model of the task and left unaddressed the way agents can learn and update their models. Humans and

other animals show robust learning of the statistical regularities of cognitive tasks, adapting their strategies and response dynamics over time. For example, during the Flanker [21] and Posner tasks [29], it is possible to learn the probability of the correct response or the probability that some cues predict the correct response (called *cue validity*) within a certain experimental block. In these tasks and many others, participants learn expectations about targets, cues or other elements, which influence their responses and movements in subsequent trials [43,19].

For these reasons, in this study we systematically investigate the learning dynamics of an embodied model of decision-making during a two-alternative forced choice task, by relying on a hybrid (discrete and continuous) active inference framework [35]. Our results show that the active inference agent can not only learn various statistical regularities of the task – namely, prior preferences for the correct target or cue validity – but also those characteristics peculiar to embodied models of cognition, i.e., the response strategies that prioritize faster or slower (but more accurate) decisions.

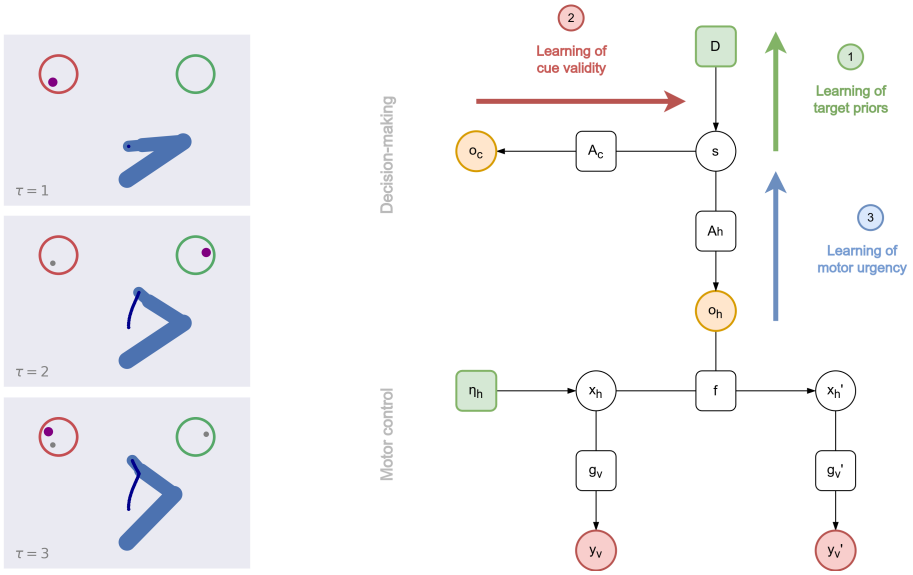
2 Methods

2.1 The two-alternative forced choice (2AFC) decision task

To study learning in embodied decisions, we designed a two-alternative forced choice (2AFC) decision task with time-varying information, i.e., wherein evidence for one choice or the other, expressed in terms of sequentially provided cues, changes throughout each trial (Figure 1a). The agent's body is a 3-DoF arm starting from a position at an equal distance from the two target buttons (red and green circles), and has to reach the target that will contain more cues (the smaller grey dots) in it. During the task, one cue after the other appears either in the left or the right circle (15 in total). The agent can move at any moment, until a certain deadline. By varying systematically the probability distributions from which the cues are sampled, we create two types of conditions: easy conditions in which cues appear in the correct target with an initial probability of 80%, which then gradually increases to 100% after 8 cues (*congruent* trials); and difficult conditions in which cues appear in the correct target with an initial probability of 20% which then increases to 100% (*incongruent* trials). Hence, during incongruent trials the cues will initially appear in the wrong target. The correct target (i.e., the circle that will contain the most cues) is sampled randomly at each trial.

The 2AFC task used here addresses the most important features of classical decision tasks (e.g., it needs making a choice based on sensory evidence, which can have different sequential statistics) but in a minimally embodied setting, as it requires a reaching movement to respond. The 2AFC task is similar to the Tokens task designed in [9], except for the fact that only a single cue at a time is visible by the agent, and that action starts from a point below the targets, not in between. Furthermore, the 2AFC task presents some analogies with the classical *Eriksen flanker task* used to analyze attention mechanisms and assess the ability to suppress inappropriate responses [12]. In the flanker task, participants observe a visual target along with congruent or incongruent cues.

The change in the probability distributions from which the cues are sampled in our task is conceptually similar to the progressive shift of attention toward the correct target typically observed in the flanker task.



(a) (b)

Fig. 1: Embodied decision setup and active inference model. (a) Experimental setup, during three consecutive discrete time steps τ. The agent controls a 3-DoF arm, which starts at a home position (blue dot) at an equal distance from the two targets (red and green circles). The current cue is displayed with a big purple dot, while the old cues are shown with smaller grey dots. For each trial, the agent has to reach the target it believes will contain more cues. (b) Factor graph of a hybrid active inference model for embodied decisions. Variables and factors are indicated by circles and squares, respectively. We highlighted three pathways related to the learning of the correct target (green arrow), cue validity (red arrow), and response strategy (blue arrow).

2.2 Hybrid active inference model for embodied decisions

To address the 2AFC task, we implemented a hybrid active inference model – shown in Figure 1b – composed of two parts: a discrete model that accumulates evidence over the cues and infers the correct target, and a continuous model that deals with the actual motor execution to reach it (see also [35]).

The discrete hidden states $\boldsymbol{s}$, encoding the probability that each target is the correct choice for the current trial, are sampled from a categorical distribution, i.e.,

$\boldsymbol{s} = Cat(\boldsymbol{D}) = \begin{bmatrix} s_{t1} & s_{t2} \end{bmatrix}$, where $\boldsymbol{D}$ are the parameters of a Dirichlet distribution and define the agent's prior beliefs. These are iteratively inferred from discrete cues $\boldsymbol{o}_c$ by inverting the cue likelihood matrix $\boldsymbol{A}_c$. This matrix takes into account some uncertainty α_c over the cues with a similar role to the *drift rate* in drift-diffusion models:

$$\boldsymbol{A}_c = \begin{bmatrix} 1-\alpha_c & \alpha_c \\ \alpha_c & 1-\alpha_c \end{bmatrix} \tag{1}$$

To compute a motor plan, the hidden states $\boldsymbol{s}$ also generate a particular combination of hand dynamics $\boldsymbol{o}_h$ through the likelihood matrix $\boldsymbol{A}_h$. Specifically, $\boldsymbol{o}_h$ encodes the probability of reaching the left target, reaching the right target, or staying, i.e., $\boldsymbol{o}_h = [o_{h,t1}, o_{h,t2}, o_{h,s}]$. A parameter α_h controls the weight of the last dynamics:

$$\boldsymbol{A}_h = \begin{bmatrix} 1-\alpha_h & 0 \\ 0 & 1-\alpha_h \\ \alpha_h & \alpha_h \end{bmatrix} \tag{2}$$

Note that the lower α_h, the less certain the agent has to be about the correct target to start moving. At each discrete step τ, a particular cue $\boldsymbol{o}_c$ and a particular hand dynamics $\boldsymbol{o}_h$ are observed and compared with the corresponding predictions. Then, the inference of the discrete hidden states $\boldsymbol{s}$ follows the equation:

$$\boldsymbol{s}_\tau = \sigma_w(k_d \ln \boldsymbol{s}_{\tau-1} + \ln \boldsymbol{A}_c^T \boldsymbol{o}_{c,\tau} + k_h \ln \boldsymbol{A}_h^T \boldsymbol{o}_{h,\tau}) \tag{3}$$

where k_d and k_h are scale parameters, and σ is a weighted softmax function, whose precision ensures fast transitions between discrete states, promoting less uncertain decisions. In all simulations, we set $k_d = 1.0$. In Equation 3, we note a prior coming from the previous step (equal to $\boldsymbol{D}$ at the beginning of the trial), a sensory likelihood for evidence accumulation, and a likelihood linked to the hand dynamics. The latter behaves as a sensory signal for the discrete model (similar to $\boldsymbol{o}_c$) and causes the agent to infer the correct target through its own movements. This *self-evidencing* mechanism of motor inference stabilizes the decision taken [1], a behavior observed in many biological scenarios [23]. See [35] for more details.

The discrete set of hand dynamics $\boldsymbol{o}_h$ is inferred via Bayesian model comparison, i.e., by comparing a prior surprise encoding the dynamics generated by the discrete model based on its guess, and a log evidence encoding the most likely hand dynamics corresponding to the current motor trajectory:

$$\boldsymbol{o}_{h,\tau} = \sigma_w(\ln \boldsymbol{A}_h \boldsymbol{s}_\tau + \int_0^T \boldsymbol{\mathcal{L}}_h dt) \tag{4}$$

For each discrete step τ, the log evidence is accumulated over a continuous period T. The derivation of Equation 4 from the free energy associated with each discrete observation can be found in [18,25]. Generating predictions about hand dynamics (as opposed to positions as in conventional hybrid models) allows the agent to interact with dynamic elements of the environment. As before, σ_w is a weighted softmax whose precision controls how fast the transition between

different dynamics occurs – e.g., high and low precisions are respectively related to abrupt and gradual movement onsets. Each hand dynamics $o_{h,m}$ is linked to a continuous dynamics function, or potential motor plan $\boldsymbol{f}_m$ in extrinsic (e.g., Cartesian) coordinates:

$$\boldsymbol{f}(\boldsymbol{\mu}_h) = \begin{bmatrix} \boldsymbol{f}_{t1} \\ \boldsymbol{f}_{t1} \\ \boldsymbol{f}_s \end{bmatrix} = \begin{bmatrix} \lambda(\boldsymbol{p}_{t1} - \boldsymbol{\mu}_h) \\ \lambda(\boldsymbol{p}_{t2} - \boldsymbol{\mu}_h) \\ \mathbf{0} \end{bmatrix} \tag{5}$$

where $\boldsymbol{\mu}_h$ is the belief over the hand position $\boldsymbol{x}_h$, λ is an attractor gain, while $\boldsymbol{p}_{t1}$ and $\boldsymbol{p}_{t2}$ are the positions of the two targets – assumed to be known and fixed. The log evidence $\mathcal{L}_{h,m}$ scores how much the mth potential dynamics is close to the belief $\boldsymbol{\mu}'_h$ over the real dynamics perceived by the agent:

$$\mathcal{L}_{h,m} = \frac{1}{2}(\boldsymbol{\mu}'^T_{h,m}\boldsymbol{p}_{x,h}\boldsymbol{\mu}'_{h,m} - \boldsymbol{f}_m(\boldsymbol{\mu}_h)^T\boldsymbol{\pi}_{x,h}\boldsymbol{f}_m(\boldsymbol{\mu}_h) \tag{6}$$

$$- \boldsymbol{\mu}'^T_h\boldsymbol{p}_{x,h}\boldsymbol{\mu}'_h + \boldsymbol{\eta}'^T_{x,h}\boldsymbol{\pi}_{x,h}\boldsymbol{\eta}'_{x,h}) \tag{7}$$

where $\boldsymbol{f}_m(\boldsymbol{\mu}_h)$ is the mth dynamics function with precision $\boldsymbol{\pi}_{x,h}$, the posterior $\boldsymbol{\mu}'_h$ encodes the estimated velocity with precision $\boldsymbol{p}_{x,h}$, and $\boldsymbol{\mu}'_{h,m}$ is the posterior of the mth dynamics. See [17,16] for more details about Bayesian model reduction, and [31,34] regarding hybrid models in dynamic contexts.

The motor plan to be realized is instead computed via Bayesian model average, i.e., by weighting each dynamics function with the respective discrete probability, i.e.,

$$\boldsymbol{\eta}'_{x,h} = \boldsymbol{o}_h \cdot \boldsymbol{f}(\boldsymbol{\mu}_h) \tag{8}$$

Hence, $\boldsymbol{\eta}'_{x,h}$ represents an average trajectory that accounts for the probability of each potential dynamics based on some discrete goal. This desired velocity enters the update of the continuous hidden states as a dynamics prediction error $\boldsymbol{\varepsilon}_{x,h} = \boldsymbol{\mu}'_h - \boldsymbol{\eta}'_{x,h}$. The continuous hidden states encode the hand position and velocity in extrinsic coordinates, and are updated – using the generalized beliefs $\tilde{\boldsymbol{\mu}}_h = [\boldsymbol{\mu}_h, \boldsymbol{\mu}'_h]$ – through the following rule:

$$\dot{\tilde{\boldsymbol{\mu}}}_h = \begin{bmatrix} \boldsymbol{\mu}'_h - \boldsymbol{\pi}_{\eta,h}\boldsymbol{\varepsilon}_{\eta,h} + \partial_{\mu_h}\boldsymbol{g}_v^T\boldsymbol{\pi}_v\boldsymbol{\varepsilon}_v + \partial_{\mu_h}\boldsymbol{f}^T\boldsymbol{\pi}_{x,h}\boldsymbol{\varepsilon}_{x,h} \\ \\ \partial_{\mu'_h}\boldsymbol{g}'^T_v\boldsymbol{\pi}'_v\boldsymbol{\varepsilon}'_v - \boldsymbol{\pi}_{x,h}\boldsymbol{\varepsilon}_{x,h} \end{bmatrix} \tag{9}$$

Here, we note two likelihood terms coming from visual observations $\boldsymbol{y}_v$ and $\boldsymbol{y}'_v$ for both orders which invert the respective generative models and keep the belief close to the actual hand trajectory; the dynamics prediction error $\boldsymbol{\varepsilon}_{x,h}$, affecting both orders either as a forward or backward message; and a prior prediction error $\boldsymbol{\varepsilon}_{\eta,h}$ that biases the belief over the hand position. The latter comes from another continuous model encoding proprioceptive trajectories (e.g., expressed in joint angles) and performing forward kinematics. As a result, the backward message automatically performs inverse kinematics, eventually driving action. See [32,30] for more details about kinematic inference.

3 Results

Here, we describe three simulations addressing different aspects of the learning dynamics of our embodied active inference agent during the 2AFC task. The three learning processes are illustrated in Figure 1b; these comprise learning the priors over the correct target (green arrow), cue validity (red arrow), and a response strategy (blue arrow). In the following, we separately analyze each of them, keeping the rest of the model parameters fixed.

3.1 Simulation 1: Learning priors over the correct target

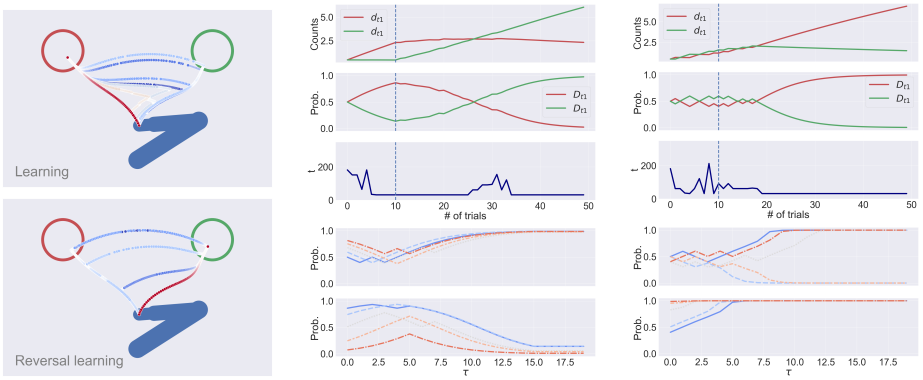
Many cognitive tasks involve learning statistical regularities; in the 2AFC task, the most common is perhaps the probability (across trials) of the correct target. Here, we simulate this learning by relying on the Dirichlet priors (or simply, counts) $\boldsymbol{d}$ over the discrete hidden states $\boldsymbol{s}$, i.e., $p(\boldsymbol{D}) = Dir(\boldsymbol{d})$ [37,11,15,40]. In active inference, learning implies that after every trial, the counts associated with each target are updated as follows:

$$\boldsymbol{d}_n = \omega \boldsymbol{d}_{n-1} + \eta \boldsymbol{s} \tag{10}$$

where n is the trial number, ω is a forgetting factor of older trials (usually initialized with a value reflecting high confidence over the prior belief), and η is the learning rate of new trials. Then, the counts are normalized to compute the priors $\boldsymbol{D}$ of the correct target for the successive trial. To assess the agent's ability to learn the target distribution, we simulate 50 *incongruent* trials, with two phases. In the initial learning phase (first 10 trials), cues appear in the left target with a starting probability of 20%, which then gradually increases to 100%. In the second reversal learning phase (subsequent 40 trials), the condition is reversed so that the correct target is the green circle. The counts $\boldsymbol{d}$ are initialized to 0.5, while the forgetting and learning rates are set to $\omega = 0.99$, $\eta = 0.2$.

The results of this simulation are shown in Figure 2b: during the early trials of the first phase (dark blue trajectories in Figure 2a), the agent moves toward the wrong direction and then changes mind. However, in later trials (dark red trajectories) it begins to move early toward the correct target, ignoring the accumulation of wrong cues. In parallel, movement onset decreases (Figure 2b). This result shows how a strong prior can overcome conflicting evidence. In the second phase, after the reversal at trial 10, the discrete prior for the left target slowly decreases, as the Dirichlet counts for the right target increase. In early trials, movement curvature increases and movement onset is slower, as the agent is uncertain about the distribution of the cues. In late trials, movement curvature decreases and movement onset fastens, as the agent learns the novel contingencies. This result shows that an embodied model can flexibly adapt to novel contingencies, solving reversal learning tasks.

Finally, Figure 2c shows statistical learning of the target priors in conditions identical to the previous simulations, but in this case the agent also exploits its motor responses to infer the correct choice, i.e., with $k_h = 0.05$ – resulting



(a) (b) (c)

Fig. 2: Results of Simulation 1. Statistical learning of the prior for the correct choice over 50 incongruent trials, composed of two phases in which the correct choice are the left and right targets, respectively. (a) Hand trajectories in equally spaced trials, during the first learning phase (top) and the second, reversal learning phase (bottom). Dark blue trajectories represent early trials, while dark red trajectories are late trials. Here, $k_h = 0$, $\alpha_c = 0.4$, and $\alpha_h = 0.4$. (b) The five panels show Dirichlet counts $\boldsymbol{d}$; discrete priors $\boldsymbol{D}$; time step of movement onset across trials; discrete hidden state s_1 (which is associated with the left target) for learning; and reversal learning, in 5 equally spaced trials over discrete time τ on a blue-to-red color scale. The vertical dashed lines indicate the time step when reversal occurs. (c) Every parameter is the same as the previous simulation, but in this case $k_h = 0.05$.

in a completely different behavior. As analyzed in [35], inferring the choice through one's own trajectories reinforces the decision taken and stabilizes the action, resulting in fewer changes of mind and more confident behavior. This generally optimizes the speed-accuracy tradeoff and decreases the risk of losing valid opportunities in a dynamic environment. However, as we show in Figure 2c, the early presentation of incongruent cues during motor inference can lead to the formation of wrong habits. In fact, the cues initially appearing in the green circle reinforce the related decision, from which a habit of reaching the wrong target gradually emerges over several trials. This result might provide a hint regarding the emergence of pathological conditions in goal-directed behavior.

3.2 Simulation 2: Learning cue validity

In active inference, not only can statistical regularities over the hidden states be learned, but also the mapping from hidden states to sensory outcomes, i.e., the likelihood matrix $\boldsymbol{A}$. This form of learning is crucial in dynamic environments, where sensory uncertainty might change depending on the context. For example, during the Posner task [29], it is common to vary the validity of the cues (e.g.,

the probability that a cue predicts the correct response) across blocks. In more mundane situations, the repeated observation of high noise on a sensory modality (e.g., turning off the lights in a room) implies that the agent has to decrease its confidence about that modality and rely more on other clues (e.g. tactile or auditory sensations), and this should appear as increased uncertainty in the related likelihood matrices. This increase (or decrease) of confidence generally occurs when the agent's predictions are repeatedly met (or violated).

The adaptation of the likelihood matrix $\boldsymbol{A}$ breaks down to keeping count of coincidences between states and outcomes. More formally, if we express $\boldsymbol{A}$ by the following Dirichlet distribution:

$$p(\boldsymbol{A}) = Dir(\boldsymbol{a}) \qquad \boldsymbol{a} = p(o_\tau|s_\tau) = \begin{bmatrix} a_{00} & \dots & a_{0i} \\ \vdots & \ddots & \vdots \\ a_{j0} & \dots & a_{ji} \end{bmatrix} \tag{11}$$

we can update the counts $\boldsymbol{a}$ similar to the learning of the prior $\boldsymbol{D}$:

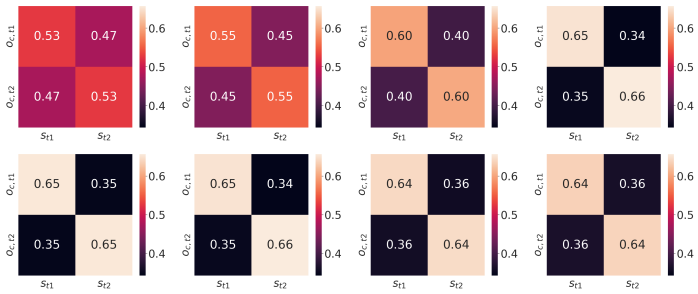
$$\boldsymbol{a}_n = \omega \boldsymbol{a}_{n-1} + \eta \sum_\tau o_\tau \otimes s_\tau \tag{12}$$

where ω and η are forgetting and learning rates, and $\otimes$ denotes the outer product (see [37] for more details). As evident, the only difference with Equation 10 is that learning is not based on the probability of the hidden states at the end of each trial, but on every occurrence of state-outcome pairs within trials. This kind of learning follows the Hebbian rule, according to which "neurons that fire together, wire together". In fact, active inference assumes that the concentration parameters can be associated with the strength of synaptic connections [15].

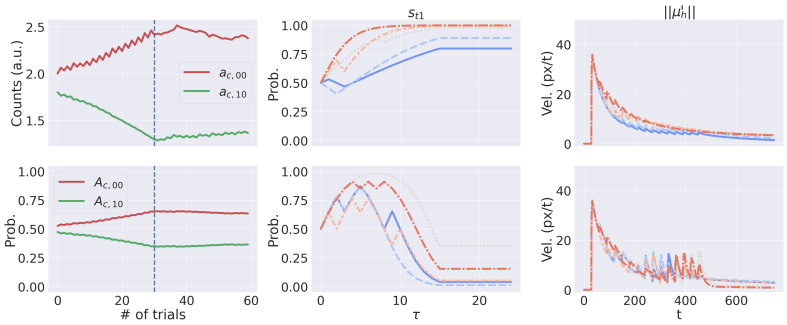
In our task, variations in uncertainty over the cue mapping become evident when the difficulty of the trials abruptly changes. To show this, we ran 30 congruent trials (equivalent to a block of the Posner task in which the cue has high validity) followed by 30 incongruent trials (equivalent to a block of the Posner task in which the cue has low validity), and analyzed the agent's behavior during learning of the likelihood matrix $\boldsymbol{A}_c$. The counts $\boldsymbol{a}_c$ are initialized with low values to provide a weak prior on evidence accumulation, $\alpha_c = 0.44$:

$$\boldsymbol{a}_c = \begin{bmatrix} 1.0 & 0.8 \\ 0.8 & 1.0 \end{bmatrix} \tag{13}$$

The forgetting and learning rates are respectively set to 0.98 and 0.01, while the parameter α_h controlling the strength of the stay dynamics is kept fixed to 0.4. Note that although we let the agent adapt to a generic form of the likelihood matrix, by sampling both targets as correct choices the initial parameterized form defined in Equation 1 is maintained (but in general this is not needed). As evident from Figure 3a, during congruent trials the parameter α_c (values on the antidiagonal) rapidly decreases as the agent's prediction of the correct target matches almost every cue – a direct consequence of the accumulation of counts shown in the first column of Figure 3b. Instead, repeated exposure of more



(a)



(b)

Fig. 3: Results of Simulation 2. (a) Learning of $\boldsymbol{A}_c$ in 30 congruent trials followed by 30 incongruent trials. The first (or second) row plots the values of $\boldsymbol{A}_c$ for 4 equally spaced trials of the congruent (or incongruent) conditions. (b) Left column: state-outcomes coincidences $\boldsymbol{a}_c$ (top), and likelihood matrix $\boldsymbol{A}_c$ (bottom), for the left target. Middle column: discrete hidden states $\boldsymbol{s}$ of left target for 4 equally spaced congruent trials (top) and incongruent trials (bottom). Right column: norm of the estimated hand velocity $\boldsymbol{\mu}'_h$ for 4 equally spaced congruent trials (top) and incongruent trials (bottom). Early-to-late trajectories are represented on a blue-to-red color scale. Note that for the sake of clarity we only plotted the dynamics associated with the left target, since the ones of the right target behave similarly.

difficult (incongruent) trials leads to a slow decrease of α_c, since the variability of the cues is much higher during the first half of the trials. This change of uncertainty reflects the rate of accumulation and hand velocity, as represented in the second and third columns of Figure 3b. In particular, the second column shows that during early congruent trials (displayed in blue), the discrete state $\boldsymbol{s}_{t1}$ associated with the left target updates slowly, never reaching complete confidence; however, during late congruent trials (displayed in red in the top panel), the update of $\boldsymbol{s}_{t1}$ is much more rapid, reflecting the increased confidence over cue validity. A specular behavior can be seen during incongruent trials (bottom

panel), during which the rapid target estimation reflects the confidence learned in the previous congruent trials, which however slowly returns to the initial rate of accumulation as repeated difficult trials are observed. Regarding hand velocity, the third column of Figure 3b shows that the agent's responses (here represented by the belief $\boldsymbol{\mu}'_h$) are gradually anticipated and increase in magnitude during congruent trials.

3.3 Simulation 3: Learning a response strategy

Besides learning the mapping between targets and cues, the agent can also learn the mapping between targets and motor responses, affecting the adopted strategy (e.g., moving faster or slower, after observing a few or several cues). Changes in response strategies are common during cognitive experiments; for example, in the Posner task, participants often slow down their responses after encountering incongruent trials [19].

Recall that the discrete hidden states generate predictions $\boldsymbol{o}_h$ related to three hand dynamics, i.e., reaching the left target, reaching the right target, or staying. In particular, the parameter α_h controls the strength of the third dynamics, and can be seen as the agent's uncertainty over the strategy to adopt: the higher α_h, the higher the probability of the correct target to initiate a movement. This is the consequence of the Bayesian model average over the three dynamics computed by Equation 8.

Learning the likelihood matrix $\boldsymbol{A}_h$ involves counting the coincidences between target probabilities and hand dynamics, via Equation 12. This implies that if the agent's hand is moving toward its target choice – as occurs more often during congruent trials – the related mapping will increase, leading to a decreased probability of the other dynamics, i.e. to stay or move to the right target. This is equivalent to a low uncertainty α_h which means that the agent's urgency to move increases. Instead, if the agent spends significant time between the two targets – as occurs more often during incongruent trials – the mapping from the two targets to the third (stay) dynamics will increase. This means a high uncertainty α_h, which translates to a low urgency to move.

To analyze this behavior, we ran a similar experiment to the previous one, i.e., 30 congruent trials followed by 30 incongruent trials. The counts $\boldsymbol{a}_h$ are initialized to:

$$\boldsymbol{a}_h = \begin{bmatrix} 2.0 & 0.0 \\ 0.0 & 2.0 \\ 1.5 & 1.5 \end{bmatrix} \tag{14}$$

such that the parameter α_h is 0.43 – corresponding to a medium urgency to move. The forgetting and learning rates are respectively set to 0.99 and 0.01, while the parameter α_c is kept fixed to 0.4. As before, the correct choice is sampled from both targets for every trial. The results of this simulation are illustrated in Figure 4. During congruent trials, the third row of $\boldsymbol{A}_h$ slightly decreases – meaning a lower strength of the stay dynamics; as shown in the top panel of the third column of Figure 4b – displaying the dynamics of the hand velocity – the

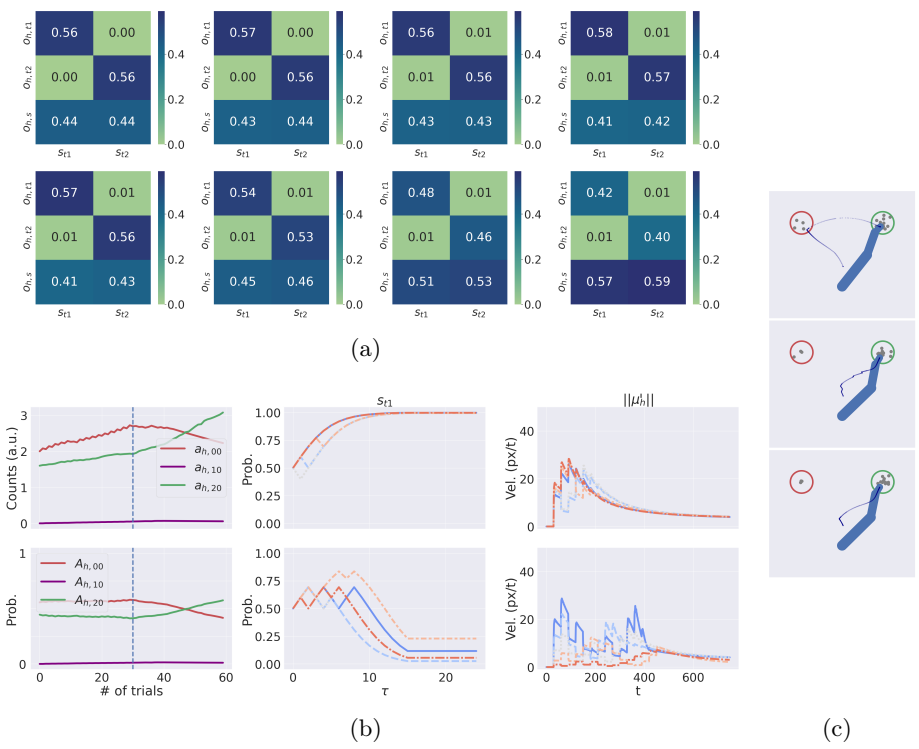


Fig. 4: Results of Simulation 3. (a) Learning of $\boldsymbol{A}_h$ in 30 congruent trials followed by 30 incongruent trials. The first (or second) row plots the values of $\boldsymbol{A}_h$ for 4 equally spaced trials of the congruent (or incongruent) trials. (b) Left column: state-outcomes coincidences $\boldsymbol{a}_h$ (top), and likelihood matrix $\boldsymbol{A}_c$ (bottom), for the left target. Middle column: discrete hidden states $\boldsymbol{s}$ of left target for 4 equally spaced congruent trials (top) and incongruent trials (bottom). Right column: norm of the estimated hand velocity $\boldsymbol{\mu}'_h$ for 4 equally spaced congruent trials (top) and congruent trials (bottom). Early-to-late trajectories are represented on a blue-to-red color scale. (c) Agent's trajectory (in dark blue) for early and late trials of incongruent conditions.

increase in the agent's confidence results in faster and stronger responses during late congruent trials (red trajectories). During incongruent trials, the learning is reversed and the last row of the matrix increases. In this case, the agent adapts its strategy to the new environmental uncertainty, and starts moving only when the correct cues are presented. Considering the bottom panel of the third column of Figure 4b, if early trials are characterized by two spikes of the estimated hand velocity $\boldsymbol{\mu}'_h$ (corresponding to the initial movement toward the wrong target and the change of mind toward the correct one) at about $t = 100$ and $t = 400$, late trials only exhibit a single spike at about $t = 500$, with a much lower magnitude. This behavior is evident from Figure 4c, showing three sample trials during the

incongruent phase; note how the agent learns a more cautious strategy, avoiding changes of mind in late trials. Finally, notice from the first column of Figure 4b that the learning of $\boldsymbol{A}_h$ mainly involves modulation of the correct reaching movements and the stay dynamics, while the probability of opposite reaching movements (values on the antidiagonal) remains low.

4 Discussion

Living organisms often face embodied decisions, which require not only selecting between outcomes but also simultaneously specifying, selecting between and executing plans to achieve these outcomes. Embodied decision models have begun to address these challenges by allowing decision and action processes to proceed in parallel and influence each other reciprocally [4,23]. A recent hybrid active inference model [35] successfully addresses embodied choices by using sensorimotor feedback from continuous dynamics – such as self-information [3] – as evidence for the decision itself.

In this study, we leverage and extend that model by studying how learning affects the agent's behavior. In particular, we focused on three kinds of learning, namely, learning target priors, cue validity, and response strategies – all of which have been reported in human cognitive studies [21,29]. In the active inference model, these three types of learning correspond to keeping Dirichlet counts of the prior matrix $\boldsymbol{D}$ encoding target probabilities, of the likelihood matrix $\boldsymbol{A}_c$ (i.e., the mapping between targets and cues), and of the likelihood matrix $\boldsymbol{A}_h$ (i.e., the mapping between targets and hand dynamics).

Taken together, our simulations show that active inference agents can dynamically optimize the enaction of embodied decisions [28] based on the observed contexts. First, by learning the statistical structure of the task, the model can form priors about the correct target, which in turn determines fast and accurate movements even in case of conflicting sensory evidence. While previous studies showed that motor inference (i.e., inferring choice alternatives from action dynamics) helps optimize the speed-accuracy trade-off during embodied decisions [23,35], here we showed that it also affects the formation of habits, which in some cases can lead to incorrect task execution. Note however that incorrect choices generally lead to negative feedback, which might help overcome this problem during cognitive experiments. Second, we showed that the embodied decision model can successfully learn the likelihood mapping between observations (cues) and correct choice alternatives (targets). In turn, this allows the model to flexibly adapt to conditions in which cues are more or less informative, as in the case of experimental blocks with different cue validity in the Posner task [29]. Third, active inference models of embodied decision can flexibly adapt their response strategy and the urgency to move to task statistics – for example, by starting movement faster (or slower) in situations where decisions require fewer (or more) cues, as exemplified by congruent (or incongruent) trials.

Finally, our results illustrate the efficacy of learning discrete preferences and strategies using sensorimotor feedback from continuous dynamics. While a few

studies addressed how to realize dynamic inference [31] and dynamic planning [33,34], learning in hybrid models of active inference is a yet unexplored topic. A promising research direction for the future would be to extend the present model with discrete transition distributions, and concurrently learn discrete and continuous dynamics, which might be key to advancing the use of active inference in tackling realistic tasks.

Acknowledgments. This research received funding from the European Union's Horizon 2020 Framework Programme for Research and Innovation under the Specific Grant Agreement No. 952215 (TAILOR); the European Research Council under the Grant Agreement No. 820213 (ThinkAhead), the Italian National Recovery and Resilience Plan (NRRP), M4C2, funded by the European Union – NextGenerationEU (Project IR0000011, CUP B51E22000150006, "EBRAINS-Italy"; Project PE0000013, CUP B53C22003630006, "FAIR"; Project PE0000006, CUP J33C22002970002 "MNESYS"), the PRIN PNRR P20224FESY, and the European Union's Horizon H2020-EIC-FETPROACT-2019 Programme for Research and Innovation under Grant Agreement 951910. The GEFORCE Quadro RTX6000 and Titan GPU cards used for this research were donated by the NVIDIA Corporation. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Buckley, C.L., Toyoizumi, T.: A theory of how active behavior stabilises neural activity: Neural gain modulation by closed-loop environmental feedback. PLoS computational biology **14**(1), e1005926 (2018)
2. Burk, D., Ingram, J.N., Franklin, D.W., Shadlen, M.N., Wolpert, D.M.: Motor effort alters changes of mind in sensorimotor decision making. PLoS ONE **9**(3), e92681 (Mar 2014). https://doi.org/10.1371/journal.pone.0092681
3. Chen, C.L., Aymanns, F., Minegishi, R., Matsuda, V.D., Talabot, N., Günel, S., Dickson, B.J., Ramdya, P.: Ascending neurons convey behavioral state to integrative sensory and action selection brain regions. Nature neuroscience **26**(4), 682–695 (2023)
4. Christopoulos, V., Schrater, P.R.: Dynamic integration of value information into a common probability currency as a theory for flexible decision making. PLoS computational biology **11**(9), e1004402 (2015)
5. Cisek, P.: Cortical mechanisms of action selection: the affordance competition hypothesis. Philosophical Transactions of the Royal Society B: Biological Sciences **362**(1485), 1585–1599 (2007)
6. Cisek, P.: Making decisions through a distributed consensus. Current opinion in neurobiology **22**(6), 927–936 (2012)
7. Cisek, P., Kalaska, J.F.: Neural mechanisms for interacting with a world full of action choices. Annual Review of Neuroscience **33**, 269–298 (2010). https://doi.org/10.1146/annurev.neuro.051508.135409
8. Cisek, P., Pastor-Bernier, A.: On the challenges and mechanisms of embodied decisions. Philosophical Transactions of the Royal Society B: Biological Sciences **369**(1655), 20130479 (2014)

9. Cisek, P., Puskas, G.A., El-Murr, S.: Decisions in changing conditions: The urgency-gating model. Journal of Neuroscience **29**(37), 11560–11571 (2009). https://doi.org/10.1523/JNEUROSCI.1844-09.2009
10. Cos, I., Pezzulo, G., Cisek, P.: Changes of mind after movement onset depend on the state of the motor system. Eneuro **8**(6) (2021)
11. Da Costa, L., Parr, T., Sajid, N., Veselic, S., Neacsu, V., Friston, K.: Active inference on discrete state-spaces: A synthesis. Journal of Mathematical Psychology **99** (2020). https://doi.org/10.1016/j.jmp.2020.102447
12. Eriksen, B.A., Eriksen, C.W.: Effects of noise letters upon the identification of a target letter in a nonsearch task. Perception and Psychophysics **16**(1), 143–149 (Jan 1974). https://doi.org/10.3758/bf03203267
13. Eriksen, C.W., Schultz, D.W.: Information processing in visual search: A continuous flow conception and experimental results. Perception & psychophysics **25**(4), 249–263 (1979)
14. Freeman, J.B., Dale, R., Farmer, T.A.: Hand in motion reveals mind in motion. Frontiers in psychology **2**, 59 (2011)
15. Friston, K., FitzGerald, T., Rigoli, F., Schwartenbeck, P., O'Doherty, J., Pezzulo, G.: Active inference and learning. Neuroscience and Biobehavioral Reviews **68**, 862–879 (Sep 2016). https://doi.org/10.1016/j.neubiorev.2016.06.022
16. Friston, K., Parr, T., Zeidman, P.: Bayesian model reduction pp. 1–32 (2018), http://arxiv.org/abs/1805.07092
17. Friston, K., Penny, W.: Post hoc Bayesian model selection. NeuroImage **56**(4), 2089–2099 (2011). https://doi.org/10.1016/j.neuroimage.2011.03.062
18. Friston, K.J., Parr, T., de Vries, B.: The graphical brain: Belief propagation and active inference **1**(4), 381–414 (2017). https://doi.org/10.1162/NETN, http://dx.doi.org/10.1162/NETN_a_00018
19. Gómez, C.M., Arjona, A., Donnarumma, F., Maisto, D., Rodríguez-Martínez, E.I., Pezzulo, G.: Tracking the time course of bayesian inference with event-related potentials: A study using the central cue posner paradigm. Frontiers in Psychology **10**, 1424 (2019)
20. Gordon, J., Maselli, A., Lancia, G.L., Thiery, T., Cisek, P., Pezzulo, G.: The road towards understanding embodied decisions. Neuroscience & Biobehavioral Reviews **131**, 722–736 (2021)
21. Gratton, G., Coles, M.G., Donchin, E.: Optimizing the use of information: strategic control of activation of responses. Journal of Experimental Psychology: General **121**(4), 480 (1992)
22. Grießbach, E., Raßbach, P., Herbort, O., Cañal-Bruland, R.: Embodied decisions during walking. Journal of Neurophysiology **128**(5), 1207–1223 (2022)
23. Lepora, N.F., Pezzulo, G.: Embodied choice: How action influences perceptual decision making. PLOS Computational Biology **11**(4), e1004110 (Apr 2015). https://doi.org/10.1371/journal.pcbi.1004110
24. Marcos, E., Cos, I., Girard, B., Verschure, P.F.: Motor cost influences perceptual decisions. PLoS One **10**(12), e0144841 (2015)
25. Parr, T., Friston, K.J.: The Discrete and Continuous Brain: From Decisions to Movement—And Back Again Thomas. Neural Computation **30**, 2319–2347 (2018). https://doi.org/10.1162/neco_a_01102
26. Parr, T., Pezzulo, G., Friston, K.J.: Active inference: the free energy principle in mind, brain, and behavior (2022)
27. Pezzulo, G., Cisek, P.: Navigating the Affordance Landscape: Feedback Control as a Process Model of Behavior and Cognition. Trends in Cognitive Sciences **20**(6), 414–424 (2016). https://doi.org/10.1016/j.tics.2016.03.013

28. Pezzulo, G., Donnarumma, F., Iodice, P., Maisto, D., Stoianov, I.: Model-based approaches to active perception and control. Entropy **19**(6) (2017). https://doi.org/10.3390/e19060266
29. Posner, M.I.: Orienting of attention. Quarterly journal of experimental psychology **32**(1), 3–25 (1980)
30. Priorelli, M., Pezzulo, G., Stoianov, I.: Active vision in binocular depth estimation: A top-down perspective. Biomimetics **8**(5) (2023). https://doi.org/10.3390/biomimetics8050445
31. Priorelli, M., Stoianov, I.: Dynamic inference by model reduction. bioRxiv (2023). https://doi.org/10.1101/2023.09.10.557043
32. Priorelli, M., Pezzulo, G., Stoianov, I.P.: Deep kinematic inference affords efficient and scalable control of bodily movements. Proceedings of the National Academy of Sciences of the United States of America **120** (2023). https://doi.org/10.1073/pnas.2309058120
33. Priorelli, M., Stoianov, I.P.: Deep hybrid models: infer and plan in the real world. arXiv (2024). https://doi.org/10.48550/arXiv.2402.10088
34. Priorelli, M., Stoianov, I.P.: Dynamic planning in hierarchical active inference. arXiv (2024). https://doi.org/10.48550/arXiv.2402.11658
35. Priorelli, M., Stoianov, I.P., Pezzulo, G.: Embodied decisions as active inference. bioRxiv (Jun 2024). https://doi.org/10.1101/2024.05.28.596181
36. Ratcliff, R., McKoon, G.: The diffusion decision model: Theory and data for two-choice decision tasks. Neural Computation **20**(4), 873–922 (Apr 2008). https://doi.org/10.1162/neco.2008.12-06-420
37. Smith, R., Friston, K.J., Whyte, C.J.: A step-by-step tutorial on active inference and its application to empirical data. Journal of Mathematical Psychology **107**, 102632 (2022). https://doi.org/10.1016/j.jmp.2021.102632
38. Song, J.H., Nakayama, K.: Hidden cognitive states revealed in choice reaching tasks. Trends in cognitive sciences **13**(8), 360–366 (2009)
39. Spivey, M.: The continuity of mind. Oxford University Press (2008)
40. Stoianov, I., Maisto, D., Pezzulo, G.: The hippocampal formation as a hierarchical generative model supporting generative replay and continual learning. Progress in Neurobiology **217**, 1–20 (2022). https://doi.org/doi.org/10.1016/j.pneurobio.2022.102329
41. Usher, M., McClelland, J.L.: The time course of perceptual choice: the leaky, competing accumulator model. Psychological review **108**(3), 550 (2001)
42. Wispinski, N.J., Gallivan, J.P., Chapman, C.S.: Models, movements, and minds: bridging the gap between decision making and action. Annals of the New York Academy of Sciences **1464**(1), 30–51 (2020)
43. Ye, W., Damian, M.F.: Effects of conflict in cognitive control: Evidence from mouse tracking. Quarterly Journal of Experimental Psychology **76**(1), 54–69 (2023)
44. Yoo, S.B.M., Hayden, B.Y., Pearson, J.M.: Continuous decisions. Philosophical Transactions of the Royal Society B **376**(1819), 20190664 (2021)

Learning in Hybrid Active Inference Models

Poppy Collis *[1,†], Ryan Singh[1,2,†], Paul F Kinghorn[1], and Christopher L Buckley[1,2]

[1] School of Engineering and Informatics, University of Sussex, Brighton, UK
{pzc20, rs773, p.kinghorn, c.l.buckley}@sussex.ac.uk,
[2] VERSES AI Research Lab, Los Angeles, California, USA

Abstract. An open problem in artificial intelligence is how systems can flexibly learn discrete abstractions that are useful for solving inherently continuous problems. Previous work in computational neuroscience has considered this functional integration of discrete and continuous variables during decision-making under the formalism of active inference [13,29]. However, their focus is on the expressive physical implementation of categorical decisions and the hierarchical mixed generative model is assumed to be known. As a consequence, it is unclear how this framework might be extended to the learning of appropriate coarse-grained variables for a given task. In light of this, we present a novel hierarchical hybrid active inference agent in which a high-level discrete active inference planner sits above a low-level continuous active inference controller. We make use of recent work in recurrent switching linear dynamical systems (rSLDS) which learn meaningful discrete representations of complex continuous dynamics via piecewise linear decomposition [22]. The representations learnt by the rSLDS inform the structure of the hybrid decision-making agent and allow us to (1) lift decision-making into the discrete domain enabling us to exploit information-theoretic exploration bonuses (2) specify temporally-abstracted sub-goals in a method reminiscent of the options framework [34] and (3) 'cache' the approximate solutions to low-level problems in the discrete planner. We apply our model to the sparse Continuous Mountain Car task, demonstrating fast system identification via enhanced exploration and successful planning through the delineation of abstract sub-goals.

Keywords: hybrid state-space models, decision-making, piecewise affine systems

1 Introduction

In a world that is inherently high-dimensional and continuous, the brain's capacity to distil and reason about discrete concepts represents a highly desirable feature in the design of autonomous systems. Humans are able to flexibly specify abstract sub-goals during planning, thereby reducing complex problems into manageable chunks [26,16]. Indeed, translating problems into discrete space offers distinct advantages in decision-making systems. For one, discrete states admit the direct implementation of classical techniques from decision theory such as dynamic programming [21]. Furthermore, we also find the computationally feasible application of information-theoretic measures (e.g. information-gain) in discrete spaces. Such measures (generally) require approximations in continuous settings but these have closed-form solutions in the discrete case [12]. While the prevailing method for translating continuous variables into discrete representations involves the simple grid-based discretisation of the state-space, this becomes extremely costly as the dimensionality increases [7,24]. We therefore seek to develop a framework which is able to smoothly handle the presence of continuous variables whilst maintaining the benefits of decision-making in the discrete domain.

* Corresponding author
- † Equal contribution

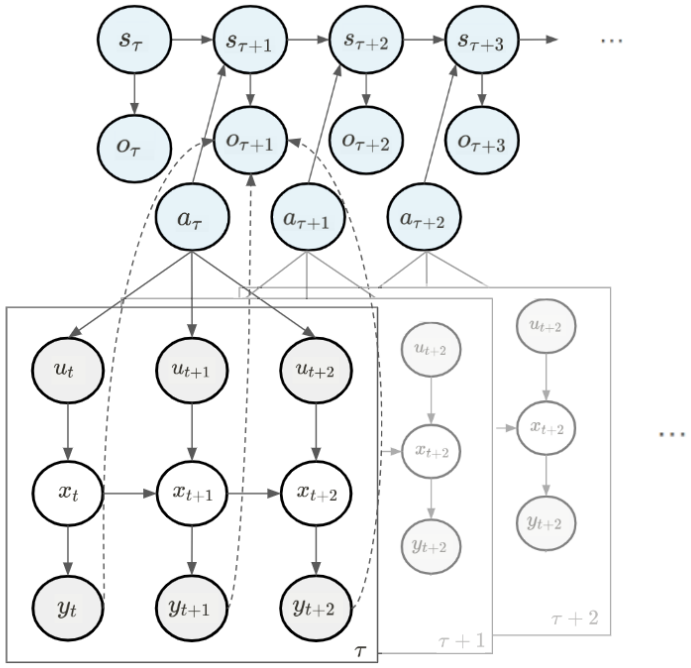


Fig. 1: **Previous discrete-continuous active inference models have focused on the physical implementation of categorical decisions in continuous space**. Here, outcomes from the high-level active inference planner select from a set of discrete models of continuous dynamics, specified by a prior over their hidden causes. This mixed generative model effectively generates discrete sequences of short continuous trajectories defined in terms of their generalised coordinates of motion. The discrete planner is formulated as a standard POMDP generative model (see Sec. 3.3) with discrete states s_τ and observations o_τ. The first action a_τ of the selected policy (see Sec. 3.2) is then passed down to the continuous active inference controller via the expected observation $q(o_{\tau+1}|a_\tau)$. This distribution weights a set of fixed point means $\{\eta_m\}_{m=0}^{M-1}$ which map the m discrete latent states into continuous state space. The resulting weighted average, $\eta_\tau = \sum_m \eta_m \cdot q(o_{\tau+1,m}|a)$, serves as the mean of a prior over hidden causes, $p(\nu) = \mathcal{N}(\eta_\tau, \pi_\tau^{-1})$, which drives the dynamics of the low-level continuous latent states $\tilde{x}_t = \{x_t, x'_t, x''_t\}$ and observations $\tilde{y}_t = \{y_t, x'_t, x''_t\}$ represented in generalised coordinates. Inherently, there is a separation of timescales in this open-loop control setup: the discrete controller sends an action down to the continuous controller which is then executed in a ballistic manner over several timesteps. After this low-level inner-loop completes, a process of Bayesian model selection is used to infer the current discrete state description of the low-level system given the trajectory of continuous observations $\tilde{y}_t$. This is then given as an observation for the discrete planner at the top. For a full treatment of this model, see [13].

1.1 Hybrid Active Inference

Here, we draw on recent work in active inference (AIF) which has foregrounded the utility of decision-making in discrete state-spaces [8,12]. Additionally, discrete AIF has been successfully combined with low-level continuous representations and used to model a range of complex behaviour including speech production, oculomotion and tool use [13,29,14,30,31]. As detailed in [13], such mixed generative models focus on the physical implementation of categorical decisions. This treatment begins with the premise that the world can be described by a set of discrete states evolving autonomously and driving the low-level

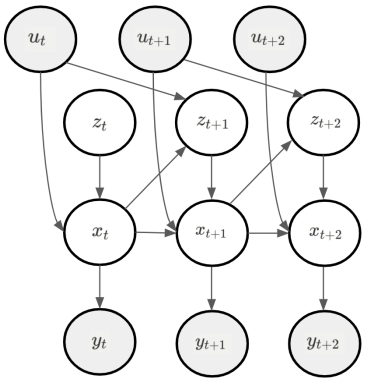


Fig. 2: **Recurrent switching linear dynamical systems (rSLDS) discover meaningful discrete states and explain how their switching behaviour depends on continuous latent states**. This class of hybrid state space model includes a recurrent dependency of the discrete latent state z_{t+1} on the continuous latent state x_t and control input u_t. As in a standard SLDS, the continuous latent dynamics are conditionally linear (dependent on the current discrete state z_t) and generate observations y_t. Note that this figure shows the recurrent-only formulation of the rSLDS (see Sec. 3.1) in which the discrete latent z_t have no dependency on z_{t-1} as is present in its canonical form.

continuous states by indexing a set of attractors (c.f. subgoals) encoded through priors which have been built into the model (see Fig. 1). While the emphasis of the above work is on mapping categorical decision-making to the continuous physical world, here, we approach the question of learning the generative model. Specifically, we seek the complete learning of appropriate discrete representations of the underlying dynamics and their manifestation in continuous space. Importantly, unlike the previous work mentioned here, we focus on instances in which the mapping between the discrete states and the continuous states is not assumed to be known. In this case, however, the assumption that higher-level discrete states autonomously drive lower-level continuous states (i.e. downward causation) becomes problematic. Any failure of the continuous system to carry out a discrete preference must be treated as an autonomous failure at the discrete level. Although useful for planning, this decoupling of the discrete from the continuous components makes it difficult to represent complex dynamics, which in turn creates difficulties in learning.

1.2 Recurrent Switching Systems

Previous work has demonstrated that models involving autonomous switching systems are often not sufficiently expressive to approximate realistic generative processes [22]. They study this problem in the context of a class of hybrid state-space model known as switching linear dynamical systems (SLDS). These models have been shown to discover meaningful behavioural modes and their causal states via the piecewise linear decomposition of complex continuous dynamics [15,11]. The authors of [22] remedy the problem associated with limited expressivity by introducing *recurrent* switching linear dynamical systems (rSLDS) (see Fig. 2). These models importantly include a dependency from the underlying continuous variables in the high-level discrete transition probabilities. By providing an understanding of the continuous latent causes of switches between the discrete states via this additional dependency, the authors demonstrate improved generative capacity and predictive performance. We propose this richer

representation can be useful for decision making and control. This recurrent transition structure can be exploited such that continuous priors can be flexibly specified for a low-level controller in order to drive the system into a desired region of the state space. Using statistical methods to fit these models not only liberates us from the need to explicitly specify a mapping between discrete and continuous states a priori, but enables effective online discovery of useful non-grid discretisations of the state-space.

1.3 Emergent descriptions for planning

Unfortunately, the inclusion of recurrent dependencies also destroys the neat separation of discrete planning from continuous control, creating unique challenges in performing roll-outs. Our central insight is to re-instate the separation by lifting the dynamical system into the discrete domain *only during planning*. We do this by approximately integrating out the continuous variables, naturally leading to spatio-temporally abstracted actions and sub-goals. Our discrete planner therefore operates purely at the level of a re-description of the discrete latents, modelling nothing of the autonomous transition probabilities but rather reflecting transitions that are possible given the discretisation of the continuous state-space. In short, we describe a novel hybrid hierarchical active inference agent [28] in which a discrete Markov decision process (MDP), informed by the representations of an rSLDS, interfaces with a continuous active inference controller implementing closed-loop control. We demonstrate the efficacy of this algorithm by applying it to the classic control task of Continuous Mountain Car [27]. We show that the exploratory bonuses afforded by the emergent discrete piecewise description of the task-space facilitates fast system identification. Moreover, the learnt representations enable the agent to successfully solve this non-trivial planning problem by specifying a series of abstract subgoals.

2 Related work

Such temporal abstractions are the focus of Hierarchical reinforcement learning (HRL), where high-level controllers provide the means for reasoning beyond the clock-rate of the low-level controllers primitive actions. [10,34,9,18]. The majority of HRL methods, however, depend on domain expertise to construct tasks, often through manually predefined subgoals as seen in [35]. Further, efforts to learn hierarchies directly in a sparse environment have typically been unsuccessful [36]. In contrast, our abstractions are a natural consequence of lifting the problem into the discrete domain and can be learnt independently of reward. In the context of control, hybrid models in the form of piecewise affine (PWA) systems have been rigorously examined and are widely applied in real-world scenarios [33,3,6]. Previous work has applied a variant on rSLDS (recurrent autoregressive hidden Markov models) to the optimal control of general nonlinear systems [2,1]. The authors use these models to the approximate expert controllers in a closed-loop behavioural cloning context. While their algorithm focuses on value function approximation, in contrast, we learn online without expert data and focus on flexible discrete planning.

3 Framework

The following sections detail the components of our Hierachical Hybrid Agent (HHA). For additional information, please refer to Appendix. A.

3.1 Generative Model: rSLDS(ro)

In the recurrent-only (ro) formulation of the rSLDS (see Fig. 2), the discrete latent states $z_t \in \{1,2,...,K\}$ are generated as a function of the continuous latents $x_t \in \mathbb{R}^M$ and the control input $u_t \in \mathbb{R}^N$ (specified by some controller) via a softmax regression model,

$$P(z_{t+1}|x_t,u_t) = softmax(W_x x_t + W_u u_t + r) \tag{1}$$

whereby $W_x \in \mathbb{R}^{K\times M}$ and $W_u \in \mathbb{R}^{K\times N}$ are weight matrices and r is a bias of size $\mathbb{R}^K$. The continuous latent states x_t evolve according to a linear dynamical system indexed by the current discrete state z_t.

$$\begin{gathered} x_{t+1}|x_t,u_t,z_t = A_{z_t}x_t + B_{z_t}u_t + b_{z_t} + \nu_t, \\ \nu_t \sim \mathcal{N}(0,Q_{z_t}) \end{gathered} \tag{2}$$

$$y_t|x_t = C_{z_t}x_t + \omega_t,\ \omega_t \sim \mathcal{N}(0,S_{z_t}) \tag{3}$$

A_{z_t} is the state transition matrix, which defines how the state x_t evolves in the absence of input. B_{z_t} is the control matrix which defines how external inputs influence the state of the system while b_{z_t} is an offset vector. At each time-step t, we observe an observation $y_t \in \mathbb{R}^M$ produced by a simple linear-Gaussian emission model with an identity matrix C_{z_t}. Both the dynamics of the continuous latents and the observations are perturbed by zero-mean Gaussian noise with covariance matrices of Q_{z_t} and S_{z_t} respectively.

Inference requires approximate methods given that the recurrent connections break conjugacy rendering the conditional likelihoods non-Gaussian. Therefore, a Laplace Variational Expectation Maximisation (EM) algorithm is used to approximate the posterior distribution over the latent variables by a mean-field factorisation into separate distributions for the discrete states $q(z)$ and the continuous states $q(x)$. The discrete state is updated via a coordinate ascent variational inference (CAVI) approach by leveraging the forward-backward algorithm. The continuous state distribution is updated using a Laplace approximation around the mode of the expected log joint probability. This involves finding the most likely continuous latent states by maximizing the expected log joint probability and computing the Hessian to approximate the posterior. Full details of the Laplace Variational EM used for learning are given in [37].

The rSLDS is initialised according to the procedure outlined in [22]. In order to learn the rSLDS parameters using Bayesian updates, conjugate matrix normal inverse Wishart (MNIW) priors are placed on the parameters of the dynamical system and recurrence weights. We learn the parameters online via observing the behavioural trajectories of the agent and updating the parameters in batches (every 1000 timesteps of the environment).

3.2 Active Inference

Equipped with a generative model, active inference specifies how an agent can solve decision making tasks [28]. Policy selection is formulated as a search procedure in which a free energy functional of predicted states is evaluated for each possible policy. Formally, we use an upper bound on the expected free energy ($\mathcal{G}$) given by:

$$\begin{aligned} \mathcal{G}_{1:T}(\pi) \leq & \underbrace{-\mathbb{E}_{Q(\mathbf{o}|\pi)}[D_{KL}[Q(\mathbf{s}|\mathbf{o},\pi)\,\|\,Q(\mathbf{s}|\pi)]]}_{\text{State Information Gain}} \\ & \underbrace{-\mathbb{E}_{Q(\mathbf{o}|\pi)}[D_{KL}[Q(\theta|\mathbf{o},\pi)\,\|\,Q(\theta|\pi)]]}_{\text{Parameter Information Gain}} \\ & \underbrace{-\mathbb{E}_{Q(\mathbf{o}|\pi)}[\ln\tilde{p}(o)]}_{\text{Utility}}. \end{aligned} \tag{4}$$

Where $\mathbf{s}=\{s_1,...,s_T\}$ and $\mathbf{o}=\{o_1,...,o_T\}$ are the states and observations being evaluated under a particular policy or sequence of actions, $\pi=a_{1:T}$. The integration of rewards in the inference procedure is achieved by biasing the agent's generative model with an optimistic prior over observing desired outcomes $\tilde{p}(o)$. Action selection then involves converting this into a probability distribution over the set of policies and sampling from this distribution accordingly.

3.3 Discrete Planner

In order to create approximate plans at the discrete level, we derive a high-level planner based on a re-description of the discrete latent states found by the rSLDS by approximately 'integrating out' the continuous variables and the continuous prior. This process involves calculating the expected free energy ($\mathcal{G}$) for a continuous controller to drive the system from one mode to another. Importantly, the structure of the lifted discrete state transition model has been constrained by the polyhedral partition of the continuous state space extracted from the parameters of the rSLDS [3]: invalid transitions are assigned zero probability while valid transitions are assigned a high probability. In order to generate the possible transitions from the rSLDS, we calculate the set of active constraints for each region from the softmax representation, $p(z\,|\,x)=\sigma(Wx+b)$. Specifically, to check that the region i is adjacent to region j, we verify the solution using a linear program,

$$-b_j = \min(W_i - W_j)x \tag{5}$$

$$\text{s.t. } (W_i - W_k)x \leq (b_i - b_k)\ \forall k \in [K] \tag{6}$$

$$\text{s.t. } x \in (x_{lb}, x_{ub}) \tag{7}$$

where (x_{lb},x_{ub}) are bounds chosen to reflect realistic values for the problem. This ensures we only lift transitions to the discrete model if they are possible. After integration, we are left with a discrete MDP which contains averaged information about all of the underlying continuous quantities. This includes information about the transitions that the structure of the task space allows, and their corresponding approximate control costs (see A.2). Note that after each batch update of the rSLDS parameters, this discrete planner must be refitted accordingly.

The lifted discrete generative model has all the components of a standard POMDP in the active inference framework:

$$P(o_{1:T},s_{1:T},A,B,\pi) = P(\pi)P(A)P(B)P(s_0)\prod_t P(s_t\,|\,s_{t-1},B,\pi)P(o_t\,|\,s_t,A) \tag{8}$$

along with prior over policies $P(\pi)=Cat(E)$, and preference distribution $\tilde{P}(o_t)=Cat(C)$. Specifically our lifted $P(\pi)$ reflects the approximate control costs of each continuous transition and $\tilde{P}(o_t)$ reflects the reward available in each mode. We assume an identity mapping between states and observation meaning the state information gain term in Eq. 4 collapses into a maximum entropy regulariser, while we maintain Dirichlet priors over the transition parameters B, facilitating directed exploration. Due to conjugate structure Bayesian updates amount to a simple count-based update of the Dirichlet parameters [25]. At each time step, the discrete planner selects a policy by sampling from the following distribution:

$$Q(\pi) = softmax(-G(\pi) + \ln P(\pi)). \tag{9}$$

[3] For a visualisation of this partitioning of the state space, see Fig. 4(a)

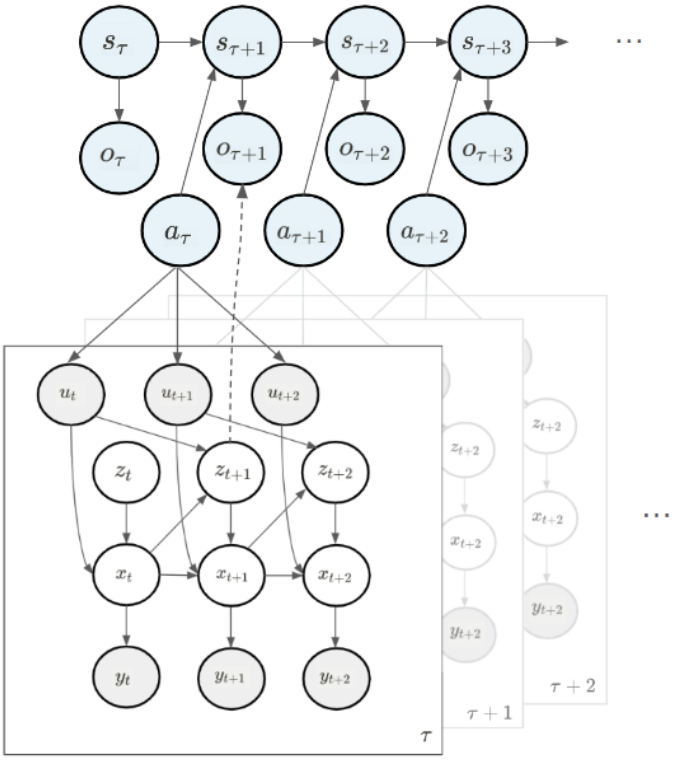


Fig. 3: **Our Hybrid Hierarchical Agent learns emergent coarse-grained descriptions of the continuous state-space for planning and control**. Like previous work on mixed generative models in active inference shown in Fig. 1, we have a discrete active inference planner sitting above a low-level continuous active inference controller. The discrete planner is constructed as a standard POMDP generative model (see Sec. 3.3) with discrete states s_τ and observations o_τ. However, our model departs from [13] in the generation of coarse-grained variables which instead emerge from the the underlying rSLDS generative model. Here, the states S of the planner are essentially a re-description of the discrete states Z found by the rSLDS. The transition model probabilities are then constrained to reflect the adjacency structure of the polyhedral partitions of the state-space found by the softmax regression component of rSLDS. The chosen action a_τ from the high-level planner selects from a discrete set of continuous active inference controllers based on both the linear dynamics of the current discrete state z_τ and a control prior for the desired next discrete state. Using the rSLDS generative model, this prior is a flexibly specified continuous point in the state-space that is in the discrete region the agent wishes to move into (see Eq. 10). Unlike the models in [13], the action a_τ is temporally abstracted with no pre-defined timescale at the lower level. Instead, the discrete planner is only re-triggered when the system enters a new discrete state (i.e. $z_t \neq z_{t-1}$). At which point, the planner observes the new discrete state z_τ of the system and constructs a plan accordingly.

The policy is then communicated to the continuous controller. Specifically, the first action of the selected policy is a requested transition $i \to j$ and is translated into a continuous control prior $\tilde{p}(x) \sim N(x_j, \Sigma_j)$

via the following link function,

$$x_j = \underset{x}{\text{argmax}} P(z=j \,|\, x,u) \tag{10}$$

whereby we numerically optimise for a point in space up to some probability threshold, T (for details on this optimisation, see A.6). These priors represents an approximately central point in the desired discrete region j requested by the action a^j. Note that these priors only need to be calculated once per refit of the rSLDS. The discrete planner infers its current state s_τ from observing z_t. Importantly, the discrete planner is only triggered when the system switches into a new mode[4]. In this sense, discrete actions are temporally abstracted and decoupled from continuous clock-time in a method reminiscent of the options framework [34].

3.4 Continuous controller

Continuous closed-loop control is handled by a set of continuous active inference controllers. For controlling the transition from mode i to mode j (x_i to x_j), the objective of the controller is to minimise the following (discrete-time) expected free energy functional [5]:

$$G_{ij}(\pi) = \mathbb{E}_{q(\cdot|x_0=x_i,\pi)}[(x_S - x_j)^T Q_f (x_S - x_j) + \sum_{t=0}^{S} u_t^T (R - \Pi_t^u) u_t] + \ln\det\Pi \tag{11}$$

[4] Or a maximum dwell-time (hyperparameter) is reached.
[5] As shown in [20] linear state space models preclude state information gain terms leaving the simplified form seen here.

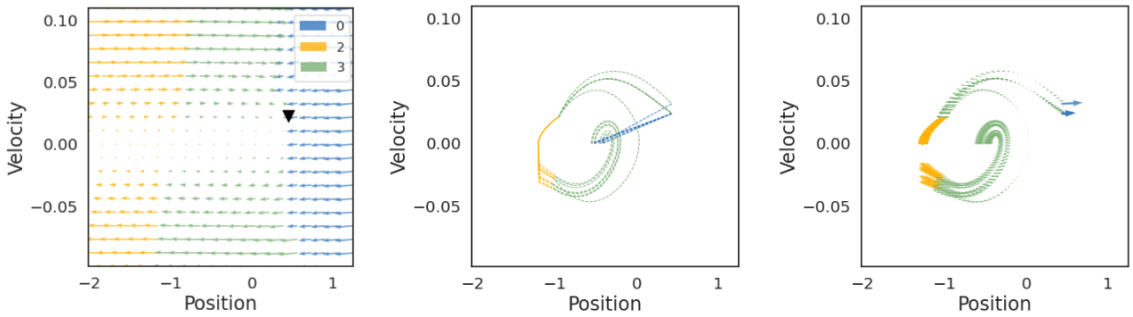


Fig. 4: **HHA solves nonlinear problems via specifying abstract sub-goals in state-space.** (a) Piecewise linear dynamics of the Continuous Mountain Car state-space found by rSLDS represented as a vector plot where magnitude of the arrows indicates how fast the state is changing at that point. Reward location shown (*black triangle*). While the rSLDS retrives 5 modes in total, here we plot only the modes seen in the position-velocity (x) space without showing the control input (u) axis. (b) Example trajectory (segments coloured by mode) showing the HHA consistently navigating to the goal. (c) Continuous control input (coloured by discrete action specified by planner and arrow size indicating magnitude and direction) over same example trajectory in (b).

Where S is the finite time horizon and the quadratic terms derive from Gaussian preferences about the final state $\tilde{p}_j(x_S) \sim N(x_j, Q_f^{-1})$ and time invariant control input prior $p(u_t) \sim N(0, R^{-1})$ (A.4). Importantly we design the control priors such that the controller only provides solutions within the environments given constraints (for further discussion, see Sec. 5). The approximate closed-loop solution to each of these sub-problems is computed offline each time the rSLDS is refitted (see A.3) using the updated parameters of the linear dynamical systems, allowing for fast discrete-only planning when online.

4 Results

To evaluate the performance of our (HHA) model, we applied it to the classic control problem of Continuous Mountain Car. This problem is particularly relevant for our purposes due to the sparse nature of the rewards, necessitating effective exploration strategies to achieve good performance. We find that the HHA finds piecewise affine approximations of the task-space and uses these discrete modes effectively to solve the task. Fig. 4 shows that while the rSLDS has divided up the space according to position, velocity and

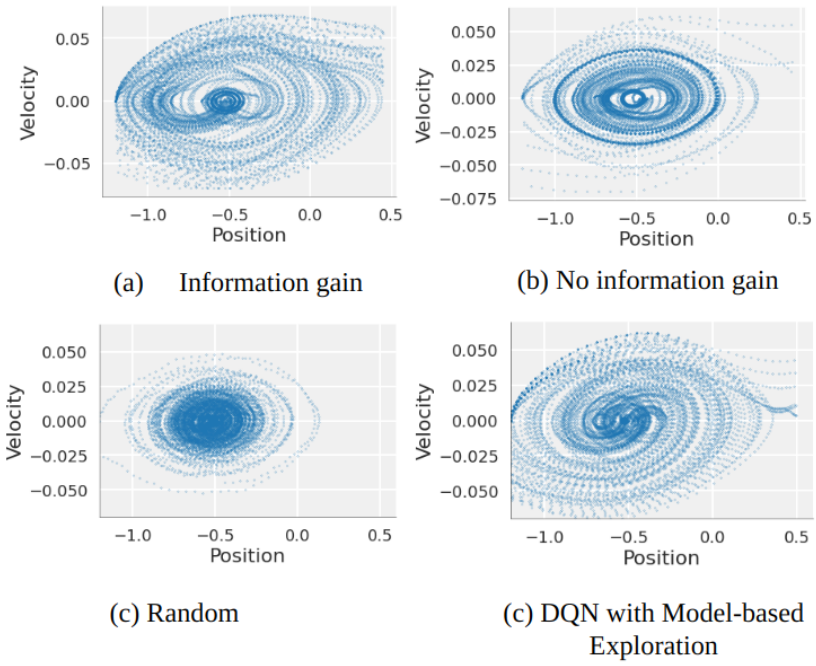


Fig. 5: **HHA with information-gain explored a wider range of the state-space**. State-space coverage in Continuous Mountain Car after 10,000 steps and best of 3 runs for (a) HHA with information-gain drive, (b) HHA without information gain drive and (c) randomly sampled continuous actions baseline. HHA with information-gain drive also shows comparable performance to (d) a Deep Q-Network with Model-Based Exploration (DQN-MBE) on the (comparably easier) Discrete Mountain Car task [17]. Exact parameters for DQN-MBE are given in Table 3 in A.8.

control input, the useful modes for solving the task are those found in the position space. Once the goal and a good approximation to the system has been found, the HHA successfully and consistently navigates to the reward. This can be seen in the example trajectories (in Fig.4b) where the agent starts at the central position [0,0] and proceeds to rock back and forth within the valley until enough momentum is gained for the car to reach the flag position at a position of 0.5. The episode terminates once the reward has been reached and the agent is re-spawned at the origin before repeating the same successful solution.

Fig. 5 shows that the HHA performs a comprehensive exploration of the state-space and significant gains in the state-space coverage are observed when using information-gain drive in policy selection compared to without. Indeed, our model competes with the state-space coverage achieved by model-based algorithms with exploratory enhancements in the discrete Mountain Car task, which is inherently easier to solve.

We compare the performance of the HHA to model-free reinforcement learning baselines (Actor-Critic and Soft Actor-Critic) and find that the HHA both finds the reward and capitalises on its experience significantly quicker than the other models (see Fig. 6). Given both the sparse nature of the task and the poor exploratory performance of random action in the continuous space, these RL baselines struggle to find the goal within 20 episodes without the implementation of reward-shaping techniques. With reference to the high sample complexity of these algorithms, our model significantly outperforms other baselines in this task.

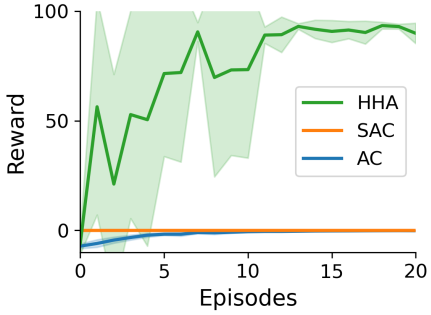


Fig. 6: **HHA both finds the reward and captilises on its experience significantly quicker than model-free RL baselines**. Average reward (+/- std) over 6 runs for Continuous Mountain Car (20 episodes, max episode length of 200 steps) for HHA (our model), Soft-Actor Critic (with 2 Q-functions), and Actor-Critic models. Note that after 20 episodes, SAC and AC are yet to find the reward and converge on a solution.

5 Discussion

The emergence of non-grid discretisations of the state-space allows us to perform fast systems identification via enhanced exploration, and successful non-trivial planning through the delineation of abstract sub-goals. Hence, the time spent exploring each region is not based on euclidean volume which helps mitigate the curse of dimensionality that other grid-based methods suffer from. Interestingly, even without information-gain, the area covered by our hybrid hierarchical agent is still notably better than that of the random continuous action control (see Fig. 5c). This is because the agent is still operating at the level of the non-grid discretisation of the state-space which acts to significantly reduce the dimensionality of the search space in a behaviourally relevant way.

Such a piecewise affine approximation of the space will incur some loss of optimality in the long run when pitted against black-box approximators. This is due to the nature of caching only approximate closed-loop solutions to control within each piecewise region, whilst the discrete planner implements open-loop control. However, this approach eases the online computational burden for flexible re-planning. Hence, in the presence of noise or perturbations within a region, the controller may adapt without any new computation. This is in contrast to other nonlinear model-based algorithms like model-predictive control where reacting to disturbances requires expensive trajectory optimisation at every step [32]. By using the piecewise affine framework, we maintain functional simplicity and interpretability through structured representation. We therefore suggest that this method is amenable to future alignment with a control-theoretic approach to safety guarantees for ensuring robust system performance and reliability. Indeed, such use of discrete approximations to continuous trajectories has been shown to improve the ability to handle uncertainty. Evidence of the efficacy of this kind of approach in machine learning applications has been exhibited in recent work by [5], which examined the problem of compounding error in imitation learning from expert demonstration. The authors demonstrated that applying a set of primitive controllers to discrete approximations of the expert trajectory effectively mitigated the accumulation of error by ensuring local stability within each chunk.

We acknowledge there may be better solutions to dealing with control input constraints than the one given in Sec. 3.4. Different approaches have been taken to the problem of implementing constrained-LQR control, such as further piecewise approximation based on defining reachability regions for the controller [4].

6 Conclusion

In summary, the successful application of our hybrid hierarchical active inference agent in the Continuous Mountain Car problem showcases the potential of recurrent switching linear dynamical systems (rSLDS) for enhancing decision-making and control in complex environments. By leveraging rSLDS to discover meaningful coarse-grained representations of continuous dynamics, our approach facilitates efficient system identification and the formulation of abstract sub-goals that drive effective planning. This method reveals a promising pathway for the end-to-end learning of hierarchical mixed generative models for active inference, providing a framework for tackling a broad range of decision-making tasks that require the integration of discrete and continuous variables. The success of our agent in this control task demonstrates the value of such hybrid models in achieving both computational efficiency and flexibility in dynamic, high-dimensional settings.

Acknowledgements This work was supported by The Leverhulme Trust through the be.AI Doctoral Scholarship Programme in Biomimetic Embodied AI. Additionally, this research received funding from the European Innovation Council via the UKRI Horizon Europe Guarantee scheme as part of the MetaTool project. We gratefully acknowledge both funding sources for their support.

Disclosure of Interests The authors have no competing interests to declare that are relevant to the content of this article.

A Appendix

A.1 Framework

Optimal Control To motivate our approximate hierarchical decomposition, we adopt the optimal control framework, specifically we consider discrete time state space dynamics of the form:

$$x_{t+1} = f(x_t, u_t, \eta_t) \tag{12}$$

with known initial condition x_0, and noise η_t drawn from some time invariant distribution $\eta_t \sim D$, where we assume f to be $p(x_{t+1} \mid x_t, u_t)$ and is a valid probability density throughout. We use $c_t : X \times U \to \mathbb{R}$ for the control cost function at time t and let $\mathbb{U}$ be the set of admissible (non-anticipative, continuous) feedback control laws, possibly restricted by affine constraints. The optimal control law for the finite horizon problem is given as:

$$J(\pi) = \mathbb{E}_{x_0,\pi}[\sum_{t=0}^{T} c_t(x_t, u_t)] \tag{13}$$

$$\pi^* = \operatorname*{argmin}_{\pi \in \mathbb{U}} J(\pi) \tag{14}$$

PWA Optimal Control The fact we do not have access to the true dynamical system f motivates the use of a piecewise affine (PWA) approximation. Also known as hybrid systems:

$$x_{t+1} = A_i x_t + B_i u_t + \epsilon_t \tag{15}$$

$$\text{when } (x_t, u_t) \in H_i \tag{16}$$

Where $\mathbb{H} = \{H_i : i \in [K]\}$ is a polyhedral partition of the space $X \times U$. In the case of a quadratic cost function, it can be shown the optimal control law for such a system is peicewise linear. Further there exist many completeness (universal approximation) type theorems for peicewise linear approximations implying if the original system is controllable, there will exist a peicewise affine approximation through which the system is still controllable [3,6].

Relationship to rSLDS We perform a canonical decomposition of the control objective J in terms of the components or modes of the system. By slight abuse of notation $[x_t = i] := [(x_t, u_t) \in H_i]$ represent the Iverson bracket.

$$J(\pi) = \sum_t \int p_\pi(x_t \mid x_{t-1}, u_t) c_t(x_t, u_t) dx_t dx_{t-1} \tag{17}$$

$$= \sum_t \int \sum_{i \in [K]} [x_{t-1} = i] p_\pi(x_t \mid x_{t-1}, u_t) c_t(x_t, u_t) dx_t dx_{t-1} \tag{18}$$

Now let z_t be the random variable on $[K]$ induced by $Z_t = i$ if $[x_t = i]$ we can rewrite the above more concisely as,

$$J(\pi) = \sum_t \int \sum_{i \in [K]} p_\pi(x_t, z_{t-1} = i \mid x_{t-1}, u_t) c_t(x_t, u_t) dx_t dx_{t-1} \tag{19}$$

$$= \sum_{i \in [K]} \sum_t \int p_\pi(x_t, z_{t-1} = i \mid x_{t-1}, u_t) c_t(x_t, u_t) dx_t dx_{t-1} \tag{20}$$

$$= \sum_{i \in [K]} \sum_t \mathbb{E}_{\pi_i}[c_t(x_t, u_t)] \tag{21}$$

which is just the expectation under a recurrent dynamical system with deterministic switches. Later (see A.5), we exploit the non-deterministic switches of rSLDS in order to drive exploration. Eq.21 demonstrates the global problem can be partitioned solving problems within each region (inner expectation), and a global discrete problem which decides which sequence of regions to visit. In the next section, we introduce a new set of variables which allows us to approximately decouple the problems.

A.2 Hierarchical Decomposition

Our aim was to decouple the discrete planning problem from the fast low-level controller. In order to break down the control objective in this manner, we first create a new discrete variable which simply tracks the transitions of z, this allows the discrete planner to function in a temporally abstracted manner.

Decoupling from clock time Let the random variable $(\zeta_s)_{s>0}$ record the transitions of $(z_t)_{t>0}$ i.e. let

$$\tau_s(\tau_{s-1}) = \min\{t : z_{t+1} \neq z_t, t > \tau_{s-1}\}, \tau_0 = 0 \tag{22}$$

be the sequence of first exit times, then ζ is given by $\zeta_s = z_{\tau_s}$. With these variables in hand, we frame a small section of the global problem as a first exit problem.

Low level problems Consider the first exit problem for exiting region i and entering j defined by:

$$\pi_{ij}(x_0) = \underset{\pi,S}{\operatorname{argmin}} J_{ij}(\pi, x_0, S) \tag{23}$$

$$J_{ij}(\pi, x_0, S) = \mathbb{E}_{\pi,x_0}[\sum_{t=0}^{S} c(x_t, u_t)] \tag{24}$$

$$\text{s.t. } (x_t, u_t) \in H_i \tag{25}$$

$$\text{s.t. } c(x,u) = 0 \text{ when } (x,u) \in \partial H_{ij} \tag{26}$$

where ∂H_{ij} is the boundary $H_i \bigcap H_j$. Due, to convexity of the polyhedral partition, the full objective admits the decomposition in terms of these subproblems,

$$J(\pi) = \sum_s J_{\zeta(s+1),\zeta(s)}(\pi, x_{t_s}, t_{s+1} - t_s) \tag{27}$$

Ideally, we would like to simply solve all possible subproblems $\{J^*_{ij}(x) : i,j \in [K] \times [K]\}$ and then find a sequence of discrete states, $\zeta(1),...,\zeta(S)$, which minimises the sum of the sub-costs, however notice each sub-cost depends on the starting state, and further this is determined by the final state of the previous problem. A pure separation into discrete and continuous problems is not possible without a simplifying assumption.

Slow and fast mode assumption The goal is to tackle the decomposed objectives individually, however the hidden constraint that the trajectories line up presents a computational challenge. Here we make the assumption that the difference in cost induced by different starting positions within a region is much less than expected difference in cost of starting in a different region. This assumption justifies using an average cost for the low-level problems to create the high-level problem.

High level problem we let $J^*_{ij} = \min_\pi \int_{x_0} J_{ij}(\pi, x_0) p(x_0)$ be the average cost of each low-level problem. We form a Markov decision process by introducing abstract actions $a \in [K]$:

$$p_{ik}(a) = \mathbb{P}(\zeta_{s+1} = k \,|\, \zeta_s = i, a = j, \pi^*_{ij}) \tag{28}$$

and let p_{π_d} be the associated distribution over trajectories induced by some discrete state feedback policy, along with the discrete state action cost $c_d(a=j,\zeta=i)=J^*_{ij}$ we may write the high level problem:

$$\pi^*_d = \min_{\pi_d} J_d(\pi,\zeta_0) \tag{29}$$

$$J_d(\pi,\zeta_0) = \mathbb{E}p_{\pi_d,\zeta_0}[\sum_{s=0}^{S} c_d(a_s,\zeta_s)] \tag{30}$$

Our overall approximate control law is then given by choosing the action of the continuous controller $\pi_{ij}(x)$ suggested by the discrete policy $\pi_d(i(x))$, or more concisely, $\pi(x)=\pi_{i(x),\pi^*_d\circ i(x)}(x)$, where i is calculates the discrete label (MAP estimate) for the continuous state x. In the next sections we describe the methods used to solve the high and low level problems.

A.3 Offline Low Level Problems: Linear Quadratic Regulator (LQR)

Rather than solve the first-exit problem directly, we formulate an approximate problem by finding trajectories that end at specific 'control priors' (see A.6). Recall the low level problem given by:

$$\pi_{ij}(x_0) = \operatorname*{argmin}_{\pi,S} J_{ij}(\pi,x_0,S) \tag{31}$$

$$J_{ij}(\pi,x_0,S) = \mathbb{E}_{\pi,x_0}[\sum_{t=0}^{S} c(x_t,u_t)] \tag{32}$$

$$\text{s.t. } (x_t,u_t)\in H_i \tag{33}$$

$$\text{s.t. } c(x,u)=0 \text{ when } (x,u)\in\partial H_{ij} \tag{34}$$

In order to approximate this problem with one solvable by a finite horizon LQR controller, we adopt a fixed goal state, $x^*\in H_j$. Imposing costs $c_t(x_t,u_t)=u_t^T R u_t$ and $c_S(x_S,u_S)=(x-x^*)Q_f(x-x^*)$. Formally we solve,

$$\pi_{ij}(x_0) = \operatorname*{argmin}_{\pi,S} J_{ij}(\pi,x_0,S) \tag{35}$$

$$J_{ij}(\pi,x_0,S) = \mathbb{E}_{\pi,x_0}[(x_S-x^*)^T Q_f (x_S-x^*) + \sum_{t=0}^{S-1} u_t^T R u_t] \tag{36}$$

$$\tag{37}$$

by integrating the discrete Ricatti equation backwards. Numerically, we found optimising over different time horizons made little difference to the solution, so we opted to instead specify a fixed horizon (hyperparameter). These solutions are recomputed offline every time the linear system matrices change.

Designing the cost matrices Instead of imposing the state constraints explicitly, we record a high cost which informs the discrete controller to avoid them. In order to approximate the constrained input we choose a suitably large control cost $R=rI$. We adopted this approach for the sake of simplicity, potentially accepting a good deal of sub-optimality. However, we believe more involved methods for solving input constrained LQR could be used in future, e.g. [3], especially because we compute these solutions offline.

A.4 Active Inference Interpretation

Expected Free Energy Here we express the fully-observed continuous (discrete time) active inference controller, without mean-field assumptions, and show it reduces to a continuous quadratic regulator. Suppose we have a linear state space model:

$$x_{t+1} = Ax_t + Bu_t + \epsilon_t \tag{38}$$

and a prior preference over trajectories $\tilde{p}(x_{1:T}) \sim N(x_T; x_f, Q_f^{-1})$, active inference specifies the agent minimises

$$G(\pi) = \mathbb{E}_{q(x_{1:T}, u_{1:T}; \pi)}[-\ln\tilde{p}(x_{1:T}, u_{1:T}) + \ln q(x_{1:T}, u_{1:T}; \pi)] \tag{39}$$

Note, since all states are fully observed we have no ambiguity term. Where $\tilde{p}(x_{1:T}, u_{1:T}) \propto \tilde{p}(x_{1:T})p(x_{1:T} \mid u_{1:T})p(u_{1:T})$, the central term is the dynamics model and the prior over controls is also gaussian, $p(u_{1:T}) = \prod_t N(u_t; 0, R^{-1})$. Finally, we adopt $q(x_{1:T}, u_{1:T}; \pi) = p(x_{1:T} \mid u_{1:T}) \prod_t \pi_t(u_t \mid x_t)$, where we parametrise the variational distributions as $\pi_t \sim N(u_t; K_t x, \Sigma_t^q)$ (where $K_t, \Sigma_t^)$ are parameters to be optimised). The expected free energy thus simplifies to:

$$G(\pi) = \mathbb{E}_{q(x_{1:T}, u_{1:T}; \pi)}[(x_T - x_F)^T Q_f (x_T - x_F) - \sum_t u_t^T (R + \Pi_t^u) u] + \ln\det\Pi \tag{40}$$

Dynamic Programming (HJB) We proceed by dynamic programming, let the 'value' function be

$$V(x_k) = \min_\pi \mathbb{E}_{q(x_{k+1:T}, u_{k:T} | x_k, \pi)}[(x_T - x_f)^T Q_f (x_T - x_f) + \sum_{t=k}^{T} u_t^T (R + \Pi_t^u) u_t] + \ln\det\Pi \tag{41}$$

As usual the value function satisfies a recursive property:

$$V(x_k) = \min_\pi \mathbb{E}_{q(x_{k+1}, u_k | x_k, \pi)}[u_k^T (R + \Pi_k^u) u_k + V(x_{k+1})] + \ln\det\Pi \tag{42}$$

We introduce the ansatz $V(x_k) = x_k^T S_k x_k$ leading to,

$$x_k^T S_k x_k = \min_\pi \mathbb{E}_{q(x_{k+1}, u_k | x_k, \pi)}[u_k^T (R + \Pi_k^u) u_k + x_{k+1}^T S_{k+1} x_{k+1}] + \ln\det\Pi \tag{43}$$

Finally we take expectations, which are available in closed form, and solve for Σ_k and K_k:

$$x_k^T S_k x_k = \min_{K_k, \Pi_k} x_k^T K_k^T R K_k x_k + tr(\Sigma_k^u (R + \Pi_t^u)) \tag{44}$$

$$+ x_k^T (A + BK_k)^T S_{k+1} (A + BK_k) x_k + tr(\Sigma_k^x S_{k+1}) + \ln\det\Pi \tag{45}$$

Solving for Σ_k and substituting,

$$\Sigma_k^q = (R + \Pi_k^u)^{-1} \tag{46}$$

$$\implies S_k = \min_{K_k} K_k^T R K_k + (A + BK_k)^T S_{k+1} (A + BK_k) \tag{47}$$

$$K_k = -(R + B^T S_{K+1} B)^{-1} B^T S_{K+1} A \tag{48}$$

Where S_k follows the discrete algebraic Riccati equation (DARE).

Thus we recover $\pi_t(u \mid x) \sim N(K_t x, \Sigma_k)$ where K_t is the traditional LQR gain, and Σ_t solves $\Sigma_k = (R + \Pi_k)^{-1}$. Here we use the deterministic maximum-a-posterori 'MAP' controller $K_t x$. However the collection of posterior variance estimates adds a different total cost depending on the variance inherent in the dynamics which can be lifted to the discrete controller.

As Belief Propagation A different perspective is as message passing: we wish to calculate the marginals $p(x_k)$ and $p(x_k,u_k)$ tilted by the preference distribution $\tilde{p}(x_k)$ and control prior $p(u)$ for this we can integrate backwards using the recursive formula

$$b(x_k) = \int b(x_k,u_k)dx_k \tag{49}$$

$$b(x_k,u_k) = \int \tilde{p}(x_k)p(x_{k+1}\,|\,x_k,u_k)p(u_k)b(x_{k+1})dx_{k+1} \tag{50}$$

from which we can extract the control law $p(u_k\,|\,x_k) = b(x_k,u_k)/b(x_k)$. To proceed we use the variational method to marginalise:

$$-\ln b(x_k) = \min_q \mathbb{E}_q[-\ln\tilde{p}(x_k,u_k)p(x_{k+1}\,|\,x_k,u_k)b(x_{k+1}) + \ln q(x_{k+1},u_k\,|\,x_k)] \tag{51}$$

making the same assumption as above about variational distributions, and introducing the ansatz $b(x_k) \sim N(x_k;0,S_k)$ leads to the same equation as 43 up to irrelevant constants.

A.5 Online high level problem

The high level problem is a discrete MDP with a 'known' model, so the usual RL techniques (approximate dynamic programming, policy iteration) apply. Here, however we choose to use a model-based algorithm with a receding horizon inspired by Active Inference, allowing us to easily incorporate exploration bonuses.

Let the Bayesian MDP be given by $\mathcal{M}_B = (S,A,P_a,R,P_\theta)$ be the MDP, where $p_a(s_{t+1}\,|\,s_t,a_t,\theta) \sim Cat(\theta_{as})$ and $p(\theta_{as}) \sim Dir(\alpha)$. We estimate the open-loop reward plus optimistic information-theoretic exploration bonuses.

Active Inference conversion We adopt the Active Inference framework for dealing with exploration. Accordingly we adopt the notation $\ln\tilde{p}(s_t,a_t) = R(s_t,a_t)$ and refer to this 'distribution' as the goal prior [23], and optimise over open loop policies $\pi = (a_0,...,a_T)$.

$$G(a_{1:T},s_0) = \mathbb{E}[\sum_{t=0}^{T} R(s_t,a_t) + IG_p + IG_s\,|\,s_0,a_{1:T}] \tag{52}$$

where parameter information-gain is given by $IG_p = D_{KL}[p_{t+1}(\theta)\,||\,p_t(\theta)]$, with $p_t(\theta) = p(\theta\,|\,s_{0:t})$. In other words, we add a bonus when we expect the posterior to diverge from the prior, which is exactly the transitions we have observed least [19].

We also have a state information-gain term, $IG_s = D_{KL}[p_{t+1}(s_{t+1})\,||\,p_t(s_{t+1})]$. In this case (fully observed), $p_{t+1}(s_{t+1}) = \delta_s$ is a one-hot vector. Leaving the term $\mathbb{E}_t[-\ln p_t(s_{t+1})]$ leading to a maximum entropy term [19].

We calculate the above with Monte Carlo sampling which is possible due to the relatively small number of modes. Local approximations such as Monte Carlo Tree Search could easily be integrated in order to scale up to more realistic problems. Alternatively, for relatively stationary environments we could instead adopt approximate dynamic programming methods for more habitual actions.

A.6 Generating continuous control priors

In order to generate control priors for the LQR controller which correspond to each of the discrete states we must find a continuous state x_i which maximises the probability of being in a desired z:

$$x_i = \underset{x}{\text{argmax}} P(z=i|x,u) \tag{53}$$

For this we perform a numerical optimisation in order to maximise this probability. Consider that this probability distribution $P(z=i|x)$ is a softmax function for the i-th class is defined as:

$$\sigma(v_i) = \frac{\exp(v_i)}{\sum_j \exp(v_j)}, v_i = w_i \cdot x + r_i \tag{54}$$

where w_i is the i-th row of the weight matrix, x is the input and r_i is the i-th bias term. The update function used in the gradient descent optimisation can be described as follows:

$$x \leftarrow x + \eta \nabla_x \sigma(v_i) \tag{55}$$

where η is the learning rate and the gradient of the softmax function with respect to the input vector x is given by:

$$\nabla_x \sigma(v_i) = \frac{\partial \sigma(v_i)}{\partial v} \cdot \frac{\partial v}{\partial x} = \sigma(v_i)(\mathbf{e}_i - \sigma(v)) \cdot W \tag{56}$$

in which $\sigma(v)$ is the vector of softmax probabilities, and $\mathbf{e}_i$ is the standard basis vector with 1 in the i-th position and 0 elsewhere. The gradient descent process continues until the probability $P(z=i|x)$ exceeds a specified threshold θ which we set to be 0.7. This threshold enforces a stopping criterion which is required for the cases in which the region z is unbounded.

A.7 Model-free RL baselines

Table 1: Summary of the Soft Actor-Critic algorithm with multiple Q-functions.

Component	Input
Q-network	3×256×256×256×2
Policy network	2×256×256×256×2
Entropy regularization coeff	0.2
Learning rates (Qnet + Polnet)	3e-4
Batchsize	60

Table 2: Summary of the Actor-Critic algorithm

Component	Input
Feature Processing	StandardScaler, RBF Kernels (4×100)
Value-network	4001 parameters (1 dense layer)
Policy network	802 parameters (2 dense layers)
Gamma	0.95
Lambda	1e-5
Learning rates (Policy + Value)	0.01

A.8 Model-based RL baseline

Table 3: Summary of DQN-MBE algorithm [17]

Component	Input
Q-network	1 hidden-layer, 48 units, ReLU
Dynamics Predictor Network (Fully Connected)	2 hidden-layers (each 24 Units), ReLU
ϵ minimum	0.01
ϵ decay	0.9995
Reward discount	0.99
Learning rates (Qnet / Dynamics-net)	0.05 / 0.02
Target Q-network update interval	8
Initial exploration only steps	10000
Minibatch size (Q-network)	16
Minibatch size (dynamics predictor network)	64
Number of recent states to fit probability model	50

References

1. Abdulsamad, H., Peters, J.: Hierarchical decomposition of nonlinear dynamics and control for system identification and policy distillation. In: Bayen, A.M., Jadbabaie, A., Pappas, G., Parrilo, P.A., Recht, B., Tomlin, C., Zeilinger, M. (eds.) Proceedings of the 2nd Conference on Learning for Dynamics and Control. Proceedings of Machine Learning Research, vol. 120, pp. 904–914. PMLR (10–11 Jun 2020)
2. Abdulsamad, H., Peters, J.: Model-based reinforcement learning via stochastic hybrid models. IEEE Open Journal of Control Systems **2**, 155–170 (2023)
3. Bemporad, A., Borrelli, F., Morari, M.: Piecewise linear optimal controllers for hybrid systems. In: Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No.00CH36334). vol. 2, pp. 1190–1194 vol.2 (2000)
4. Bemporad, A., Morari, M., Dua, V., Pistikopoulos, E.N.: The explicit linear quadratic regulator for constrained systems. Automatica **38**(1), 3–20 (2002)
5. Block, A., Jadbabaie, A., Pfrommer, D., Simchowitz, M., Tedrake, R.: Provable guarantees for generative behavior cloning: Bridging low-level stability and high-level behavior (2023)
6. Borrelli, F., Bemporad, A., Fodor, M., Hrovat, D.: An mpc/hybrid system approach to traction control. IEEE Transactions on Control Systems Technology **14**(3), 541–552 (2006)
7. Coulom, R.: Efficient selectivity and backup operators in monte-carlo tree search. In: van den Herik, H.J., Ciancarini, P., Donkers, H.H.L.M.J. (eds.) Computers and Games. pp. 72–83. Springer Berlin Heidelberg, Berlin, Heidelberg (2007)
8. Da Costa, L., Parr, T., Sajid, N., Veselic, S., Neacsu, V., Friston, K.: Active inference on discrete state-spaces: A synthesis. Journal of Mathematical Psychology **99**, 102447 (2020)
9. Daniel, C., van Hoof, H., Peters, J., Neumann, G.: Probabilistic inference for determining options in reinforcement learning. Machine Learning **104**(2), 337–357 (Sep 2016)
10. Dayan, P., Hinton, G.E.: Feudal reinforcement learning. In: Hanson, S., Cowan, J., Giles, C. (eds.) Advances in Neural Information Processing Systems. vol. 5. Morgan-Kaufmann (1992)
11. Fox, E., Sudderth, E., Jordan, M., Willsky, A.: Nonparametric bayesian learning of switching linear dynamical systems. In: Koller, D., Schuurmans, D., Bengio, Y., Bottou, L. (eds.) Advances in Neural Information Processing Systems. vol. 21. Curran Associates, Inc. (2008)
12. Friston, K., Da Costa, L., Tschantz, A., Kiefer, A., Salvatori, T., Neacsu, V., Koudahl, M., Heins, R., Sajid, N., Markovic, D., Parr, T., Verbelen, T., Buckley, C.: Supervised structure learning (12 2023)
13. Friston, K.J., Parr, T., de Vries, B.: The graphical brain: belief propagation and active inference. Network neuroscience **1**(4), 381–414 (2017)
14. Friston, K.J., Sajid, N., Quiroga-Martinez, D.R., Parr, T., Price, C.J., Holmes, E.: Active listening. Hearing research **399**, 107998 (2021)
15. Ghahramani, Z., Hinton, G.E.: Variational Learning for Switching State-Space Models. Neural Computation **12**(4), 831–864 (04 2000)
16. Gobet, F., Lane, P., Croker, S., Cheng, P., Jones, G., Oliver, I., Pine, J.: Chunking mechanisms in human learning. Trends in cognitive sciences **5**, 236–243 (07 2001)
17. Gou, S.Z., Liu, Y.: DQN with model-based exploration: efficient learning on environments with sparse rewards. CoRR **abs/1903.09295** (2019)
18. Hafner, D., Lee, K.H., Fischer, I., Abbeel, P.: Deep hierarchical planning from pixels (2022)
19. Heins, C., Millidge, B., Demekas, D., Klein, B., Friston, K., Couzin, I., Tschantz, A.: pymdp: A python library for active inference in discrete state spaces. arXiv preprint arXiv:2201.03904 (2022)
20. Koudahl, M.T., Kouw, W.M., de Vries, B.: On Epistemics in Expected Free Energy for Linear Gaussian State Space Models. Entropy **23**(12), 1565 (Dec 2021)
21. LaValle, S.M.: Planning Algorithms, chap. 2. Cambridge University Press, Cambridge (2006)
22. Linderman, S.W., Miller, A.C., Adams, R.P., Blei, D.M., Paninski, L., Johnson, M.J.: Recurrent switching linear dynamical systems (2016)
23. Millidge, B., Tschantz, A., Seth, A.K., Buckley, C.L.: On the relationship between active inference and control as inference (2020)

24. Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M.A., Fidjeland, A.K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., Hassabis, D.: Human-level control through deep reinforcement learning. Nature **518**, 529–533 (2015)
25. Murphy, K.P.: Machine learning: a probabilistic perspective. MIT press (2012)
26. Newell, A., Simon, H.A.: Human Problem Solving. Prentice-Hall, Englewood Cliffs, NJ (1972)
27. OpenAI: Continuous mountain car environment (2021), accessed: 2024-05-25
28. Parr, T., Pezzulo, G., Friston, K.: Active Inference: The Free Energy Principle in Mind, Brain, and Behavior. MIT Press (2022)
29. Parr, T., Friston, K.J.: The discrete and continuous brain: from decisions to movement—and back again. Neural computation **30**(9), 2319–2347 (2018)
30. Parr, T., Friston, K.J.: The computational pharmacology of oculomotion. Psychopharmacology **236**(8), 2473–2484 (2019)
31. Priorelli, M., Stoianov, I.P.: Hierarchical hybrid modeling for flexible tool use (2024)
32. Schwenzer, M., Ay, M., Bergs, T., Abel, D.: Review on model predictive control: an engineering perspective. The International Journal of Advanced Manufacturing Technology **117**(5), 1327–1349 (Nov 2021)
33. Sontag, E.: Nonlinear regulation: The piecewise linear approach. IEEE Transactions on Automatic Control **26**(2), 346–358 (1981)
34. Sutton, R.S., Precup, D., Singh, S.: Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. Artificial Intelligence **112**(1), 181–211 (1999)
35. Tessler, C., Givony, S., Zahavy, T., Mankowitz, D., Mannor, S.: A deep hierarchical approach to lifelong learning in minecraft. Proceedings of the AAAI Conference on Artificial Intelligence **31**(1) (Feb 2017)
36. Vezhnevets, A.S., Osindero, S., Schaul, T., Heess, N., Jaderberg, M., Silver, D., Kavukcuoglu, K.: FeUdal networks for hierarchical reinforcement learning. In: Precup, D., Teh, Y.W. (eds.) Proceedings of the 34th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 70, pp. 3540–3549. PMLR (06–11 Aug 2017)
37. Zoltowski, D.M., Pillow, J.W., Linderman, S.W.: Unifying and generalizing models of neural dynamics during decision-making (2020)

Planning to avoid ambiguous states through Gaussian approximations to non-linear sensors in active inference agents

Wouter M. Kouw[0000−0002−0547−4817]

Bayesian Intelligent Autonomous Systems laboratory
TU Eindhoven, Eindhoven, the Netherlands
w.m.kouw@tue.nl

Abstract. In nature, active inference agents must learn how observations of the world represent the state of the agent. In engineering, the physics behind sensors is often known reasonably accurately and measurement functions can be incorporated into generative models. When a measurement function is non-linear, the transformed variable is typically approximated with a Gaussian distribution to ensure tractable inference. We show that Gaussian approximations that are sensitive to the curvature of the measurement function, such as a second-order Taylor approximation, produce a state-dependent ambiguity term. This induces a preference over states, based on how accurately the state can be inferred from the observation. We demonstrate this preference with a robot navigation experiment where agents plan trajectories.

Keywords: Active inference · Free energy minimization · Bayesian filtering · Non-linear sensing · Control systems · Planning · Navigation

1 Introduction

In nature, intelligent agents build a model to infer the causes of their sensations [2]. In engineering, we are able to utilize knowledge of the relevant physics to structure such a model. In particular, we often know how sensors measure states of the world. For example, we know how radar measures relative velocity and distance [19]. Measurement functions that are non-linear transformations of state variables pose challenges to state estimation, which are often dealt with using Gaussian approximations of the transformed variables [8,14]. We show that for certain Gaussian approximations, an active inference agent will prefer to avoid states because it already knows that state estimation will be difficult.

Active inference agents are based on free energy functionals that rank policies on explorative and goal-directed behaviour [5,7,6,16]. The expected free energy functional can be understood through its decomposition into a cross-entropy term between states and observations given action ("ambiguity"), and a Kullback-Leibler divergence between the posterior predictive and a goal prior distribution ("risk") [7,18,4]. We show that Gaussian approximations of a non-linear observation function that are itself linear in the covariance matrix, e.g.,

first-order Taylor and the unscented transform [9], lead to ambiguity terms that are constant over states. This echoes an earlier finding that agents with a linear Gaussian state-space model exhibit a constant ambiguity term [10]. However, utilizing a second-order Taylor approximation induces a non-constant ambiguity term. Under this model, the agent will avoid states where the non-linear measurement function curves strongly. Our contributions are:

- Analysis of ambiguity in expected free energy functions under three different Gaussian approximations.
- An experiment where a robot must plan a trajectory and navigate to a goal prior distribution, testing the effect of the ambiguity term.

2 Problem statement

We want to plan a trajectory for a robot across a plane. The robot's state at time k is its planar position and time derivatives, $x_k \in \mathbb{R}^{D_x}$. The robot does not sense position directly, but has to infer it from noisy measurements $y_k \in \mathbb{R}^{D_y}$, produced by a sensor through a non-linear mapping $g : \mathbb{R}^{D_x} \to \mathbb{R}^{D_y}$ and measurement noise $v_k \in \mathbb{R}^{D_y}$. It accepts control inputs $u_k \in \mathbb{R}^{D_u}$ and moves according to linear dynamics with a transition matrix $A \in \mathbb{R}^{D_x \times D_x}$, control matrix $B \in \mathbb{R}^{D_x \times D_u}$ and process noise $e_k \in \mathbb{R}^{D_x}$. Overall, we consider robot systems described with discrete-time state-space models of the form:

$$x_k = Ax_{k-1} + Bu_k + e_k\,, \qquad e_k \sim \mathcal{N}(0, Q)\,, \tag{1}$$

$$y_k = g(x_k) + v_k\,, \qquad v_k \sim \mathcal{N}(0, R)\,, \tag{2}$$

where $Q \in \mathbb{R}_+^{D_x \times D_x}$, $R \in \mathbb{R}_+^{D_y \times D_y}$ are noise covariance matrices.

The goal is to find a sequence of T controls $\bar{u}_k = u_{k+1}, \ldots u_{k+T}$ that produces future states close to a desired state x_*. Agents must plan every time-step. The challenge is that errors in state estimation may cause drastic changes in the planned trajectory, which can lead an agent astray.

Example Consider a robot with position and velocity states that must move from position $x_0 = (0, -1)$ to $x_* = (0, 1)$. Its state transition, control and process noise covariance matrices are given by:

$$A = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},\; B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix},\; Q = \begin{bmatrix} \sigma_1^2 \frac{\Delta t^3}{3} & 0 & \sigma_1^2 \frac{\Delta t^2}{2} & 0 \\ 0 & \sigma_2^2 \frac{\Delta t^3}{3} & 0 & \sigma_2^2 \frac{\Delta t^2}{2} \\ \sigma_1^2 \frac{\Delta t^2}{2} & 0 & \sigma_1^2 \Delta t & 0 \\ 0 & \sigma_2^2 \frac{\Delta t^2}{2} & 0 & \sigma_2^2 \Delta t \end{bmatrix}, \tag{3}$$

for $\Delta t = 0.5$, $\sigma_1 = \sigma_2 = 0.1$. Measurements are produced by a sensor station at $(0, 0)$ that reports relative angle $\phi_k \in [-\pi\ \pi]$ and relative distance $d_k \in [0, \infty)$. The mapping and measurement noise covariance matrix are:

$$g(x_k) = \begin{bmatrix} \phi_k \\ d_k \end{bmatrix} = \begin{bmatrix} \sqrt{x_{1k}^2 + x_{2k}^2} \\ \arctan(x_{1k}, x_{2k}) \end{bmatrix}, \qquad R = \begin{bmatrix} \rho_1^2 & 0 \\ 0 & \rho_2^2 \end{bmatrix}, \tag{4}$$

where $\rho_1 = \rho_2 = 0.001$. Suppose it uses an extended Kalman filter (first-order Taylor approximation) for state estimation and a finite-horizon model-predictive control objective of the form:

$$J_k(\bar{u}_k) = \sum_{t=k+1}^{k+T} \big((A\hat{x}_{t-1}+Bu_t) - x_*\big)^\intercal C\big((A\hat{x}_{t-1}+Bu_t) - x_*\big) + \eta u_t^2\,, \qquad (5)$$

where $\hat{x}$ is the mean state, $\hat{x}_t = A\hat{x}_{t-1} + Bu_t$, C is a cost matrix (ones for position, zeros for velocity) and η a regularization parameter. Minimizing this objective every time-step produces the control sequence $\bar{u}_k^{\text{MPC}} = \arg\min J_k(\bar{u}_k)$. Such an agent will first plan a trajectory moving directly forward, as described in Figure 1 (left). However, as it approaches the sensor station, its state estimate become progressively more inaccurate and it makes increasingly more drastic adjustments to the control plan (see $k = 5$ in Figure 1 middle). Figure 1 (right) shows the executed trajectory over a trial of 10 steps, demonstrating that the agent lost track of the robot's state and did not successfully reach the target.

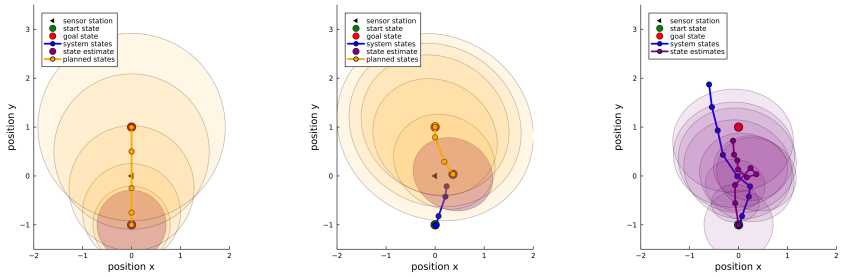


Fig. 1. (Left) Planned trajectory at $k = 1$, from start to goal directly over the sensor station. (Middle) Planned trajectory at $k = 5$ showing a mismatch between true and estimated state resulting in a strong adjustment to the planned trajectory. (Right) Executed trajectory over a trial of 10 steps demonstrates the agent losing track of the robot when it approaches the sensor station.

3 Agent specification

3.1 Probabilistic Model

The agent's model will have Gaussian prior distributions over states and controls,

$$p(x_0) = \mathcal{N}(x_0 \mid m_0, S_0)\,, \qquad \mathcal{N}(u_k \mid 0, \eta^{-1}I)\,, \qquad (6)$$

with mean m_0, covariance matrix S_0, precision η and identity matrix I. The agent's state transition will also be expressed as a Gaussian distribution:

$$p(x_k \mid x_{k-1}, u_k) = \mathcal{N}(x_k \mid Ax_{k-1} + Bu_k, Q)\,. \qquad (7)$$

Let the marginal state x_k be Gaussian distributed, i.e., $p(x_k) = \mathcal{N}(x_k \,|\, m_k, S_k)$. We restrict our attention to approximations of the nonlinear sensor $g(x_k)$ that produce Gaussian joint distributions over states and observations [14], i.e.,

$$p(y_k, x_k) \approx \mathcal{N}\Big(\begin{bmatrix} x_k \\ y_k \end{bmatrix} \;|\; \begin{bmatrix} m_k \\ \mu_k \end{bmatrix}, \begin{bmatrix} S_k & \Gamma_k \\ \Gamma_k^\intercal & \Sigma_k \end{bmatrix} \Big) \,. \tag{8}$$

From the joint, we obtain a conditional distribution of observations given states:

$$p(y_k \,|\, x_k) \approx \mathcal{N}\big(y_k \,|\, \mu_k + \Gamma_k^\intercal S_k^{-1}(x_k - m_k), \Sigma_k - \Gamma_k S_k^{-1} \Gamma_k^\intercal\big) \,. \tag{9}$$

This distribution is linear in x_k, and will allow for exact Bayesian filtering. But note that the parameters μ_k, Γ_k and Σ_k may be nonlinear functions of x_k, depending on the type of Gaussian approximation (specifics treated in Section 4), and may be a richer representation of the effect of $g(x_k)$.

3.2 Inferring states

We assume that, when inferring states, the agent has observed the system output $y_k = \hat{y}_k$ and input $u_k = \hat{u}_k$. Let $\mathcal{D}_k \triangleq \{\hat{y}_i, \hat{u}_i\}_{i=1}^k$ refer to data observed thus far. Given the known executed control, state estimation follows the general Bayesian filtering equations [14]. Firstly, the prior predictive distribution is given by:

$$p(x_k|\, \hat{u}_k, \mathcal{D}_{k\text{-}1}) = \int p(x_k|x_{k\text{-}1}, \hat{u}_k)\, p(x_{k\text{-}1}|\mathcal{D}_{k\text{-}1})\, \mathrm{d}x_{k\text{-}1} = \mathcal{N}(x_k|\, \bar{m}_k, \bar{S}_k) \,. \tag{10}$$

with $\bar{m}_k \triangleq Am_{k-1} + B\hat{u}_k$ and $\bar{S}_k \triangleq AS_{k-1}A^\intercal + Q$. This prediction is corrected by the observation through Bayes' rule [14],

$$p(x_k \,|\, \mathcal{D}_k) = \frac{p(\hat{y}_k \,|\, x_k)}{p(\hat{y}_k \,|\, \mathcal{D}_{k\text{-}1})} p(x_k \,|\, \hat{u}_k, \mathcal{D}_{k\text{-}1}) = \mathcal{N}(x_k \,|\, m_k, S_k) \,, \tag{11}$$

with $m_k = \bar{m}_k + \Gamma_k \Sigma_k^{-1}(\hat{y}_k - \mu_k)$ and $S_k = \bar{S}_k - \Gamma_k \Sigma_k^{-1} \Gamma_k$.

3.3 Inferring controls

We will discuss the inference procedure first for a single step into the future, and then generalize to a finite horizon of length T. Predictions for the future state and observation are made by unrolling the generative model to $t = k + 1$:

$$p(y_t, x_t, u_t \,|\, \mathcal{D}_k) = p(y_t \,|\, x_t) p(x_t \,|\, u_t; \mathcal{D}_k) p(u_t) \,. \tag{12}$$

We will use an expected free energy functional to infer a posterior distribution over the control u_t [13]:

$$\mathcal{F}_k[q] = \int q(y_t \,|\, x_t) \int q(x_t, u_t) \ln \frac{q(x_t, u_t)}{p(y_t, x_t, u_t \,|\, \mathcal{D}_k)} \mathrm{d}(u_t, x_t) \mathrm{d}y_t \,. \tag{13}$$

The variational model is specified to be:

$$q(y_t \mid x_t) \triangleq p(y_t \mid x_t)\,, \qquad q(x_t, u_t) \triangleq p(x_t \mid u_t; \mathcal{D}_k) q(u_t)\,. \tag{14}$$

Constraining $q(y_t \mid x_t)$ to the Gaussian approximation defined in Eq. 9 allows us to study deterministic approximations in an expected free energy minimization context. Given this variational model, Eq. 13 may be re-arranged to:

$$\begin{aligned}
\mathcal{F}_k[q] &= \int p(y_t \mid x_t) \int p(x_t \mid u_t; \mathcal{D}_k) q(u_t) \ln \frac{p(x_t \mid u_t; \mathcal{D}_k) q(u_t)}{p(y_t, x_t, u_t; \mathcal{D}_k)} \mathrm{d}(x_t, u_t) \mathrm{d}y_t & (15)\\
&= \int q(u_t) \Big(\int p(y_t, x_t \mid u_t; \mathcal{D}_k) \ln \frac{p(x_t \mid u_t; \mathcal{D}_k) q(u_t)}{p(y_t, x_t \mid u_t; \mathcal{D}_k) p(u_t)} \mathrm{d}(y_t, x_t) \Big) \mathrm{d}u_t & (16)\\
&= \int q(u_t) \Big(\ln \frac{q(u_t)}{p(u_t)} + \underbrace{\int p(y_t, x_t | u_t; \mathcal{D}_k) \ln \frac{p(x_t \mid u_t; \mathcal{D}_k)}{p(y_t, x_t | u_t; \mathcal{D}_k)} \mathrm{d}(y_t, x_t)}_{\mathcal{J}_k(u_t)} \Big) \mathrm{d}u_t\,. & (17)
\end{aligned}$$

We refer to $\mathcal{J}_k(u_t)$ as the expected free energy *function* as it depends on the value of u_t not on its distribution. Under $\mathcal{J}_k(u_t) = \ln(1/\exp(-\mathcal{J}_k(u_t)))$, the expected free energy functional can be concisely expressed as:

$$\mathcal{F}_k[q] = \int q(u_t) \ln \frac{q(u_t)}{p(u_t) \exp\left(-\mathcal{J}_k(u_t)\right)} \mathrm{d}u_t\,. \tag{18}$$

The above is a Kullback-Leibler divergence, which is minimal when

$$q^*(u_t) \propto p(u_t) \exp\left(-\mathcal{J}_k(u_t)\right)\,. \tag{19}$$

The proportionality is due to the implicit constraint[1] that $q^*(u_t)$ should integrate to 1. To work out the expectation in Eq. 17, we first decompose the joint over states and observations into

$$p(y_t, x_t \mid u_t; \mathcal{D}_k) = p(x_t \mid y_t, u_t; \mathcal{D}_k) p(y_t)\,, \tag{20}$$

and then intervene on the marginal distribution over y_t with a distribution reflecting desired future observations (a.k.a. goal prior) [11]:

$$p(y_t) \to p(y_t \mid y_*) = \mathcal{N}(y_t \mid \mu_*, \Sigma_*)\,. \tag{21}$$

The next step involves applying Bayes' rule in the inverse direction:

$$\frac{1}{p(x_t \mid y_t, u_t; \mathcal{D}_k)} = \frac{p(y_t \mid u_t; \mathcal{D}_k)}{p(y_t \mid x_t) p(x_t \mid u_t; \mathcal{D}_k)}\,, \tag{22}$$

[1] A more rigorous treatment would define a Lagrangian with normalization and marginalization constraints [17]. However, such a treatment is inconsequential when resorting to MAP estimation, as will be pursued later in the paper.

where the marginal prediction for the future observation is:

$$p(y_t \mid u_t; \mathcal{D}_k) = \int p(y_t \mid x_t)\, p(x_t \mid u_t; \mathcal{D}_k) \mathrm{d}x_t \tag{23}$$

$$= \int \mathcal{N}(\begin{bmatrix} x_t \\ y_t \end{bmatrix} \mid \begin{bmatrix} \bar{m}_t \\ \mu_t \end{bmatrix}, \begin{bmatrix} \bar{S}_t & \Gamma_t \\ \Gamma_t^\intercal & \Sigma_t \end{bmatrix}) \mathrm{d}x_t = \mathcal{N}(y_t \mid \mu_t, \Sigma_t)\,. \tag{24}$$

Note that μ_t and Σ_t depend on u_t through $\bar{m}_t$. Plugging Eqs. 21 and 22 into Eq. 20 yields:

$$\mathcal{J}_k(u_t) = \int p(y_t, x_t | u_t; \mathcal{D}_k) \ln \frac{p(x_t|u_t; \mathcal{D}_k)}{p(y_t \mid y_*)} \frac{p(y_t \mid u_t; \mathcal{D}_k)}{p(y_t \mid x_t) p(x_t \mid u_t; \mathcal{D}_k)} \mathrm{d}(y_t, x_t) \tag{25}$$

$$= \underbrace{\int p(y_t, x_t \mid u_t; \mathcal{D}_k) \Big[-\ln \frac{p(y_t, x_t \mid u_t; \mathcal{D}_k)}{p(x_t \mid u_t; \mathcal{D}_k)} \Big] \mathrm{d}(y_t, x_t)}_{\text{ambiguity}}$$

$$+ \underbrace{\int \Big[\int p(y_t, x_t \mid u_t; \mathcal{D}_k) \mathrm{d}x_t \Big] \ln \frac{p(y_t \mid u_t; \mathcal{D}_k)}{p(y_t \mid y_*)} \mathrm{d}y_t}_{\text{risk}}\,. \tag{26}$$

"Risk" refers to the Kullback-Leibler (KL) divergence between predicted and desired future observations. The inner integral in the risk term leads to a Gaussian distribution (Eq. 24) and the KL divergence between Gaussians is [3]:

$$\mathbb{E}_{p(y_t \mid u_t; \mathcal{D}_k)} \Big[\ln \frac{p(y_t \mid u_t; \mathcal{D}_k)}{p(y_t \mid y_*)} \Big] = \frac{1}{2} \Big(\ln \frac{|\Sigma_*|}{|\Sigma_t|} - D_y + \mathrm{tr}\big(\Sigma_*^{-1}(\Sigma_t + \Psi_*)\big) \Big)\,. \tag{27}$$

where $\Psi_* \triangleq (\mu_* - \mu_t)(\mu_* - \mu_t)^\intercal$. "Ambiguity" refers to the conditional entropy of the future observations given the future states.

Lemma 1. *Ambiguity, as defined in Eq. 26, for a generative model described in Eq. 12 and a variational distribution described in Eq. 14, is:*

$$\mathbb{E}_{p(y_t, x_t | u_t; \mathcal{D}_k)} \big[-\ln \frac{p(y_t, x_t | u_t; \mathcal{D}_k)}{p(x_t | u_t; \mathcal{D}_k)} \big] = \frac{D_y}{2} \ln(2\pi e) + \frac{1}{2} \ln |\Sigma_t - \Gamma_t^\intercal \bar{S}_t^{-1} \Gamma_t|. \tag{28}$$

The proof is in Appendix A. Note that the first term does not depend on the state x_t. Plugging Eqs. 27 and 28 into the expected free energy function (Eq. 26) produces:

$$\mathcal{J}_k(u_t) = \text{constants} + \frac{1}{2} \mathrm{tr}\big(\Sigma_*^{-1}(\Sigma_t + \Psi_*)\big) + \frac{1}{2} \ln \frac{\big|\Sigma_t - \Gamma_t^\intercal \bar{S}_t^{-1} \Gamma_t\big|}{\big|\Sigma_t\big|}\,. \tag{29}$$

Note that Γ_t, Σ_t and Ψ_* depend on u_t. The above steps can be generalized to a longer time horizon $t = k{+}1, \ldots k{+}T$. The prior is independent over time, so the joint factorizes as: $p(\bar{u}) = \prod_{t=1}^T p(u_t)$ (see Eq. 6). This means that the expected free energy function over $\bar{u}_k$ also factorizes to a sum of recursive expected free energy functions $\mathcal{J}_k(\bar{u}_k) = \sum_{t=1}^T \mathcal{J}_k(u_t)$, where the predicted state distribution parameters $\bar{m}_t, \bar{S}_t$ are updated through the state transition (Eq. 10).

We are interested in the most probable value under the approximate control posterior, i.e., the MAP estimate:

$$\hat{u} = \arg\max_{\bar{u}\in\mathcal{U}} q^*(\bar{u}) = \arg\min_{\bar{u}\in\mathcal{U}} \sum_{t=k+1}^{k+T} \mathcal{J}_t(u_t) - \ln p(u_t)\,, \tag{30}$$

where $\mathcal{U} \subset \mathbb{R}^T$ is the space of affordable controls over T steps. Constraints such as motor force limits can be imposed during optimization.

4 Gaussian approximations

We discuss the three most popular Gaussian approximations to non-linear transformations of Gaussian random variables: the first and second-order Taylor series approximations (used in extended Kalman filters) and the unscented transform (used in the unscented Kalman filter) [9,8][14, Ch. 5].

The first-order Taylor series approximation effectively linearizes the non-linear observation function $g(x_t)$. Since ambiguity is known to be constant over states under a linear observation function [10], it is no surprise that the first-order Taylor also leads to an ambiguity term that is constant over states.

Theorem 1. *Let $G_x(\bar{m}_t)$ be the Jacobian of g with respect to x_t, evaluated at $\bar{m}_t$. Under a* first-*order Taylor approximation, the parameters Σ_t, Γ_t are:*

$$\Sigma_t = G_x(\bar{m}_t)\bar{S}_t G_x(\bar{m}_t)^\intercal + R\,, \qquad \Gamma_t = \bar{S}_t G_x(\bar{m}_t)^\intercal\,. \tag{31}$$

With these parameters, the ambiguity term does not depend on the state x_t:

$$\mathbb{E}_{p(y_t,x_t \mid u_t;\mathcal{D}_k)}\big[\ln \frac{p(x_t \mid u_t;\mathcal{D}_k)}{p(y_t,x_t \mid u_t;\mathcal{D}_k)}\big] = -\frac{1}{2}\ln|R|\,. \tag{32}$$

The proof is in Appendix B. Perhaps surprisingly, under the second-order Taylor approximation, the ambiguity term varies as a function of the state x_t.

Theorem 2. *Let $G^{(i)}_{xx}(\bar{m}_t)$ be the Hessian of the i-th element of the non-linear observation function evaluated at $\bar{m}_t$, and let e_i be a canonical basis vector. The parameters Σ_t, Γ_t computed through a* second-*order Taylor approximation are:*

$$\begin{aligned}\Sigma_t &= G_x(\bar{m}_t)\bar{S}_t G_x(\bar{m}_t)^\intercal + \frac{1}{2}\sum_{i=1}^{D_y}\sum_{j=1}^{D_y} e_i e_j^\intercal tr\big(G^{(i)}_{xx}(\bar{m}_t)\bar{S}_t G^{(j)}_{xx}(\bar{m}_t)\bar{S}_t\big) + R\\ \Gamma_t &= \bar{S}_t G_x(\bar{m}_t)^\intercal\,.\end{aligned} \tag{33}$$

With these parameters, the ambiguity term depends on x_t through:

$$\begin{aligned}\mathbb{E}_{p(y_t,x_t \mid u_t;\mathcal{D}_k)}\big[\ln \frac{p(x_t \mid u_t;\mathcal{D}_k)}{p(y_t,x_t \mid u_t;\mathcal{D}_k)}\big] &= \\ -\frac{1}{2}\ln\Big|\frac{1}{2}\sum_{i=1}^{D_y}\sum_{j=1}^{D_y} e_i e_j^\intercal tr\big(G^{(i)}_{xx}(\bar{m}_t)\bar{S}_t G^{(j)}_{xx}(\bar{m}_t)\bar{S}_t\big) + R\Big|\,.\end{aligned} \tag{34}$$

The proof is in Appendix C.

Interestingly, the ambiguity is constant over x_t for the unscented transform.

Theorem 3. *Define $2D_x + 1$ sigma points as:*

$$\chi_0 \triangleq \bar{m}_t, \quad \chi_i \triangleq \bar{m}_t + \sqrt{D_x + \lambda}\big[\sqrt{\bar{S}_t}\big]_i, \quad \chi_{D_x+i} \triangleq \bar{m}_t - \sqrt{D_x + \lambda}\big[\sqrt{\bar{S}_t}\big]_i, \quad (35)$$

where $i = 1, \dots D_x$, $[\cdot]_i$ denotes the i-th column of a matrix, and $\sqrt{S}$ denotes the matrix square root such that $\sqrt{S}\sqrt{S} = S$. The parameter $\lambda \triangleq \alpha^2(D_x + \kappa) - D_x$ depends on free parameters α and κ. Define $2D_x + 1$ weights as:

$$w_0 \triangleq \frac{\lambda}{D_x + \lambda} + (1 - \alpha^2 + \beta)\,, \qquad w_i \triangleq \frac{1}{D_x + \lambda}\,, \quad (36)$$

for $i = 1, \dots 2D_x$ and β as an additional free parameter. Under these sigma points and weights, the parameters μ_t, Σ_t, Γ_t are [14, Eq. 5.89]:

$$\mu_t = \frac{\lambda}{D_x+\lambda} g(\chi_0) + \sum_{i=1}^{2D_x} \frac{1}{2(D_x+\lambda)} g(\chi_i)\,, \quad \Gamma_t = \sum_{i=0}^{2D_x} w_i(\chi_i - \bar{m}_t)(g(\chi_i) - \mu_t)^\intercal\,,$$

$$\Sigma_t = \sum_{i=0}^{2D_x} w_i(g(\chi_i) - \mu_t)(g(\chi_i) - \mu_t)^\intercal + R\,. \quad (37)$$

Then, the ambiguity is independent of the state:

$$\mathbb{E}_{p(y_t, x_t \mid u_t; \mathcal{D}_k)}\big[\ln \frac{p(x_t \mid u_t; \mathcal{D}_k)}{p(y_t, x_t \mid u_t; \mathcal{D}_k)}\big] = -\frac{1}{2}\ln|R|\,. \quad (38)$$

The proof can be found in the Appendix D. This result is conjectured to hold for other Gaussian approximations that are linear in their estimate of the covariance matrix, for example the Gauss-Hermite approximation [14, Ch. 6].

5 Experiments

Our experiment is as described in Section 2, with the nonlinear observation function $g(\cdot)$ measuring relative angle and distance to a base station. Examples of sensors include Hall effect and ultrasound sensors. The robot starts at $x_0 =$ [0 -1 0 0] and must reach $x_* =$ [0 1 0 0]. The agent's state prior distribution's parameters were $m_0 =$ [0 -1 0 0] and $S_0 = 0.5I$. Its control prior precision was set to a tiny value, $\eta = 1.0 \cdot 10^{-8}$, so as to best study the effects of ambiguity and risk. It was given a goal prior of $m_* = g(x_*)$ and $S_* = 0.5I$.

We will compare three agents[2]: firstly, an agent that uses the first-order Taylor approximation, referred to as EFE1. Secondly, an agent with a second-order Taylor approximation, referred to as EFE2. Thirdly, an agent with a second-order Taylor approximation but with only the risk term included, referred to as EFER. The difference between EFER and EFE2 reflects the effect of the ambiguity term, while the difference between EFE1 and EFE2 reflects the effect of

[2] Details and code at: https://github.com/biaslab/IWAI2024-ambiguity

the second-order Gaussian approximation. Figure 2 plots the value of the control objective function at every position in state-space, under a state covariance matrix of $S_t = I$. States close to the sensor station are red and will lead to high values under the control objective. Note that the area around the sensor station increases from EFE1 to EFER due to the curvature of the relative distance sensor. The white markers are the approximate minimizers for this choice of S_t matrix. Comparing EFER and EFE2, we can see that ambiguity increases the cost of being close to the sensor station.

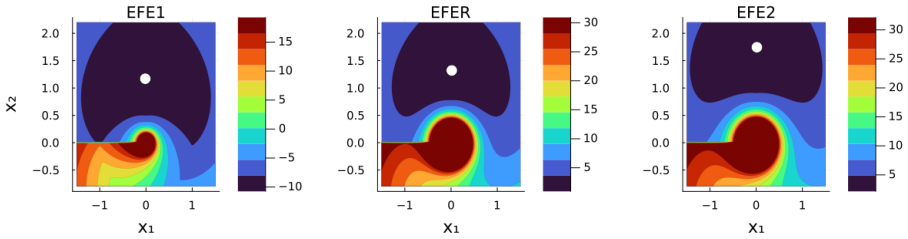


Fig. 2. Value under three EFE functions over a plane: EFE1 is risk and ambiguity under a first-order Taylor approximation, EFER is risk only under a second-order Taylor approximation and EFE2 is both risk and ambiguity under a second-order Taylor approximation. White markers indicate minimizers. Note that each EFE function induces a different preference over states.

We ran 100 Monte Carlo experiments. Figure 3 plots the average trajectory of $T = 30$ steps taken by the EFE1, EFER and EFE2 agents. Ribbons indicate the standard error of the mean at every time-point. Note that all agents avoid the sensor station, with EFE2 taking the widest curve (EFE1 and EFER turn at $x_1 = 1.0$ while EFE2 turns at $x_1 = 1.5$). EFE1 and EFER lose track of the robot in a number of experiments (like the model predictive controller in Sec. 2), leading to a more volatile average trajectory. EFE2 has the smoothest average trajectory, indicating that the ambiguity term helps planning.

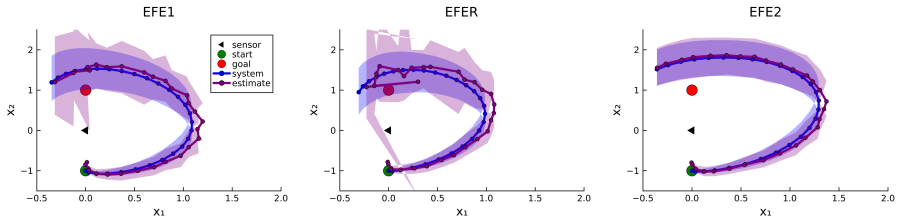


Fig. 3. Trajectories of agents under three EFE functions, averaged over 100 Monte Carlo samples (ribbon is standard deviation of the mean). The robot starts at the green marker and must reach the red goal marker. All agents avoid the sensor station, with EFE2 taking the widest curve and having the smoothest average trajectory.

6 Discussion

One could argue that our analysis is more about model selection than inference, as each Gaussian approximation essentially constitutes a different generative model. In that sense, the experiments only indicate that richer approximations of nonlinear functions lead to better performance, which is not surprising. However, the result is more subtle than that since the unscented transform is richer than the first-order Taylor (produces a more accurate mean estimate [8]) but apparently still leads to constant ambiguity. No, the approximation must be sensitive to how the covariance between states and observations changes as a function of g's curvature. It would be interesting to extend this work with parameter estimation, such as inferring the process noise covariance matrix using a Wishart distribution [15], or the state transition matrix with a Matrix-Normal distribution [1,12].

7 Conclusion

We examined active inference agents with linear Gaussian distributed dynamics and a non-linear measurement function. We found that the first-order Taylor series and unscented transform approximations to the non-linearly transformed states lead to expected free energy functions with ambiguity terms that are constant over states. A second-order Taylor approximation leads to a state-dependent ambiguity term, inducing a preference over states.

Acknowledgments. The author gratefully acknowledges financial support from the Eindhoven Artificial Intelligence Systems Institute (EAISI) at TU Eindhoven.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

A Appendix: proof of Lemma 1

Proof. The cross-entropy is split into two entropies that simplify according to:

$$
\begin{aligned}
&\mathbb{E}_{p(y_t,x_t \mid u_t;\mathcal{D}_k)}\big[-\ln \frac{p(y_t,x_t \mid u_t;\mathcal{D}_k)}{p(x_t \mid u_t;\mathcal{D}_k)}\big] \\
&= -\int \mathcal{N}\big(\begin{bmatrix}x_t\\ y_t\end{bmatrix} \mid \begin{bmatrix}\bar{m}_t\\ \mu_t\end{bmatrix}, \begin{bmatrix}\bar{S}_t & \Gamma_t\\ \Gamma_t^\intercal & \Sigma_t\end{bmatrix}\big) \ln \mathcal{N}\big(\begin{bmatrix}x_t\\ y_t\end{bmatrix} \mid \begin{bmatrix}\bar{m}_t\\ \mu_t\end{bmatrix}, \begin{bmatrix}\bar{S}_t & \Gamma_t\\ \Gamma_t^\intercal & \Sigma_t\end{bmatrix}\big)\, \mathrm{d}(y_t,x_t) \\
&\quad + \int \mathcal{N}(x_t \mid \bar{m}_t, \bar{S}_t) \ln \mathcal{N}(x_t \mid \bar{m}_t, \bar{S}_t) \mathrm{d}x_t \qquad (39)\\
&= \frac{D_x + D_y}{2}\ln(2\pi e) + \frac{1}{2}\ln\Big|\begin{bmatrix}\bar{S}_t & \Gamma_t\\ \Gamma_t^\intercal & \Sigma_t\end{bmatrix}\Big| - \frac{D_x}{2}\ln(2\pi e) + \frac{1}{2}\ln|\bar{S}_t| \qquad (40)\\
&= \frac{D_y}{2}\ln(2\pi e) + \frac{1}{2}\ln\big(|\bar{S}_t| \cdot |\Sigma_t - \Gamma_t^\intercal \bar{S}_t^{-1}\Gamma_t|\big) - \frac{1}{2}\ln|\bar{S}_t| \qquad (41)\\
&= \frac{D_y}{2}\ln(2\pi e) + \frac{1}{2}\ln|\Sigma_t - \Gamma_t^\intercal \bar{S}_t^{-1}\Gamma_t|\,. \qquad (42)
\end{aligned}
$$

B Appendix: proof of Theorem 1

Proof. Plugging Σ_t, Γ_t from (31) into the result from Lemma 1, yields:

$$-\frac{1}{2}\ln|\Sigma_t - \Gamma_t^\intercal \bar{S}_t^{-1}\Gamma| \tag{43}$$
$$= -\frac{1}{2}\ln\left|G_x(\bar{m}_t)\bar{S}_t G_x(\bar{m}_t)^\intercal + R - G_x(\bar{m}_t)\bar{S}_t^\intercal \bar{S}_t^{-1}\bar{S}_t G_x(\bar{m}_t)^\intercal\right|.$$

Since the covariance matrix $\bar{S}_t$ is symmetric, $\bar{S}_t^\intercal \bar{S}_t^{-1} = \bar{S}_t\bar{S}_t^{-1} = I$. Thus:

$$-\frac{1}{2}\ln\left|G_x(\bar{m}_t)\bar{S}_t G_x(\bar{m}_t)^\intercal + R - G_x(\bar{m}_t)\bar{S}_t G_x(\bar{m}_t)^\intercal\right| = -\frac{1}{2}\ln|R|. \tag{44}$$

C Appendix: proof of Theorem 2

Proof. Plugging Σ_t, Γ_t from (33) into the result from Lemma 1, yields:

$$-\frac{1}{2}\ln\Big|\,\Sigma_t - \Gamma_t^\intercal \bar{S}_t^{-1}\Gamma\,\Big| = -\frac{1}{2}\ln\Big|\,G_x(\bar{m}_t)\bar{S}_t G_x(\bar{m}_t)^\intercal + \tag{45}$$
$$\frac{1}{2}\sum_{i=1}^{D_y}\sum_{j=1}^{D_y} e_i e_j^\intercal \mathrm{tr}\big(G_{xx}^{(i)}(\bar{m}_t)\bar{S}_t G_{xx}^{(j)}(\bar{m}_t)\bar{S}_t\big) + R - G_x(\bar{m}_t)\bar{S}_t^\intercal\bar{S}_t^{-1}\bar{S}_t G_x(\bar{m}_t)^\intercal\Big|$$
$$= -\frac{1}{2}\ln\Big|\,\frac{1}{2}\sum_{i=1}^{D_y}\sum_{j=1}^{D_y} e_i e_j^\intercal \mathrm{tr}\big(G_{xx}^{(i)}(\bar{m}_t)\bar{S}_t G_{xx}^{(j)}(\bar{m}_t)\bar{S}_t\big) + R\,\Big|. \tag{46}$$

The covariance matrix $\bar{S}_t$ is symmetric. Thus, $\bar{S}_t^\intercal\bar{S}_t^{-1} = \bar{S}_t\bar{S}_t^{-1} = I$. Note that the Hessian $G_{xx}^{(i)}(\bar{m}_t)$ depends on the inferred mean of the predicted state $\bar{m}_t$.

D Appendix: proof of Theorem 3

Proof. Plugging Σ_t, Γ_t from (37) into the result from Lemma 1, gives:

$$-\frac{1}{2}\ln\big|\Sigma_t - \Gamma_t^\intercal\bar{S}_t^{-1}\Gamma\big| = -\frac{1}{2}\ln\Big|\sum_{i'=0}^{2D_x} w_{i'}(g(\chi_{i'})-\mu_t)(g(\chi_{i'})-\mu_t)^\intercal + R$$
$$-\Big(\sum_{i=0}^{2D_x} w_i(\chi_i-\bar{m}_t)(g(\chi_i)-\mu_t)^\intercal\Big)^\intercal \bar{S}_t^{-1}\Big(\sum_{j=0}^{2D_x} w_j(\chi_j-\bar{m}_t)(g(\chi_j)-\mu_t)^\intercal\Big)\Big|. \tag{47}$$

The second term can be re-arranged to:

$$\Big(\sum_{i=0}^{2D_x} w_i(\chi_i-\bar{m}_t)(g(\chi_i)-\mu_t)^\intercal\Big)^\intercal \bar{S}_t^{-1}\Big(\sum_{j=0}^{2D_x} w_j(\chi_j-\bar{m}_t)(g(\chi_j)-\mu_t)^\intercal\Big)$$
$$= \sum_{i=0}^{2D_x}\sum_{j=0}^{2D_x} w_i(g(\chi_i)-\mu_t)(\chi_i-\bar{m}_t)^\intercal\bar{S}_t^{-1}w_j(\chi_j-\bar{m}_t)(g(\chi_j)-\mu_t)^\intercal. \tag{48}$$

Note that for $j=0$, $(\chi_j-\bar{m}_t)=(\bar{m}_t-\bar{m}_t)=0$. Let $D_\lambda = D_x+\lambda$. For $j\geq 1$:

$$(\chi_i-\bar{m}_t)^\intercal \bar{S}_t^{-1} w_j (\chi_j-\bar{m}_t) \tag{49}$$

$$= (\bar{m}_t+(-1)^i\sqrt{D_\lambda}\big[\sqrt{\bar{S}_t}\big]_i-\bar{m}_t)^\intercal \bar{S}_t^{-1}\frac{1}{D_\lambda}(\bar{m}_t+(-1)^j\sqrt{D_\lambda}\big[\sqrt{\bar{S}_t}\big]_j-\bar{m}_t)$$

$$= \frac{1}{D_\lambda}(\sqrt{D_\lambda})^2(-1)^{i+j}\big[\sqrt{\bar{S}_t}\big]_i^\intercal \bar{S}_t^{-1}\big[\sqrt{\bar{S}_t}\big]_j \tag{50}$$

$$= (-1)^{i+j}\big[\sqrt{\bar{S}_t}\big]_i^\intercal \bar{S}_t^{-1}\big[\sqrt{\bar{S}_t}\big]_j . \tag{51}$$

Column selection $\big[\cdot\big]_i$ is equivalent to right-multiplication with a canonical basis vector e_i;

$$\big[\sqrt{\bar{S}_t}\big]_i^\intercal \bar{S}_t^{-1}\big[\sqrt{\bar{S}_t}\big]_j = \big(\sqrt{\bar{S}_t}e_i\big)^\intercal \bar{S}_t^{-1}\big(\sqrt{\bar{S}_t}e_j\big) = e_i^\intercal \sqrt{\bar{S}_t}^\intercal \bar{S}_t^{-1}\sqrt{\bar{S}_t}e_j . \tag{52}$$

Since $\bar{S}_t$ is a normal matrix, the eigendecomposition $\bar{S}_t = V\Omega V^{-1}$ generates an orthonormal eigenvector matrix V, implying $V^{-1}=V^\top$, and a diagonal matrix of eigenvalues Ω. This means that $\sqrt{\bar{S}_t}=V\Omega^{1/2}V^{-1}$, and that:

$$\sqrt{\bar{S}_t}^\intercal \bar{S}_t^{-1}\sqrt{\bar{S}_t} = \big(V\Omega^{1/2}V^{-1}\big)^\intercal (V\Omega V^{-1})^{-1}\big(V\Omega^{1/2}V^{-1}\big) \tag{53}$$

$$= V\Omega^{1/2}V^{-1}V\Omega^{-1}V^{-1}V\Omega^{1/2}V^{-1} \tag{54}$$

$$= VV^{-1} = I. \tag{55}$$

Therefore, $e_i^\intercal I e_j$ will be 1 for all $i=j$ and 0 for $i\neq j$. We can thus identify two cases in the double sum in (48), one of which is always 0:

$$\sum_{i=0}^{2D_x}\sum_{j=0}^{2D_x} w_i(g(\chi_i)-\mu_t)(\chi_i-\bar{m}_t)^\intercal \bar{S}_t^{-1} w_j(\chi_j-\bar{m}_t)(g(\chi_j)-\mu_t)^\intercal \tag{56}$$

$$= \sum_{i=0}^{2D_x}\sum_{j=i} w_i(g(\chi_i)-\mu_t)(-1)^{(i+j)}\,1\,(g(\chi_j)-\mu_t)^\intercal$$

$$+ \sum_{i=0}^{2D_x}\sum_{j\neq i} w_i(g(\chi_i)-\mu_t)(-1)^{(i+j)}\,0\,(g(\chi_j)-\mu_t)^\intercal \tag{57}$$

$$= \sum_{i=0}^{2D_x} w_i(g(\chi_i)-\mu_t)(g(\chi_i)-\mu_t)^\intercal , \tag{58}$$

where the $(-1)^{(i+j)}$ drops out because for $i=j$, $i+j$ will always be even. One may now recognize that (47) has two terms that cancel each other:

$$-\frac{1}{2}\ln|\sum_{i'=0}^{2D_x} w_{i'}(g(\chi_{i'})-\mu_t)(g(\chi_{i'})-\mu_t)^\intercal + R - \sum_{i=0}^{2D_x} w_i(g(\chi_i)-\mu_t)(g(\chi_i)-\mu_t)^\intercal|$$

$$= -\frac{1}{2}\ln|R| . \tag{59}$$

This concludes the proof.

References

1. Barber, D., Chiappa, S.: Unified inference for variational Bayesian linear Gaussian state-space models. Advances in Neural Information Processing Systems **19** (2006)
2. Conant, R.C., Ross Ashby, W.: Every good regulator of a system must be a model of that system. International Journal of Systems Science **1**(2), 89–97 (1970)
3. Cover, T.M.: Elements of information theory. John Wiley & Sons (1999)
4. Da Costa, L., Parr, T., Sajid, N., Veselic, S., Neacsu, V., Friston, K.: Active inference on discrete state-spaces: A synthesis. Journal of Mathematical Psychology **99**, 102447 (2020)
5. Friston, K.: The free-energy principle: a unified brain theory? Nature Reviews Neuroscience **11**(2), 127–138 (2010)
6. Friston, K., FitzGerald, T., Rigoli, F., Schwartenbeck, P., Pezzulo, G.: Active inference: a process theory. Neural Computation **29**(1), 1–49 (2017)
7. Friston, K., Rigoli, F., Ognibene, D., Mathys, C., Fitzgerald, T., Pezzulo, G.: Active inference and epistemic value. Cognitive Neuroscience **6**(4), 187–214 (2015)
8. Gustafsson, F., Hendeby, G.: Some relations between extended and unscented Kalman filters. IEEE Transactions on Signal Processing **60**(2), 545–555 (2011)
9. Julier, S.J., Uhlmann, J.K.: Unscented filtering and nonlinear estimation. Proceedings of the IEEE **92**(3), 401–422 (2004)
10. Koudahl, M.T., Kouw, W.M., de Vries, B.: On epistemics in expected free energy for linear Gaussian state space models. Entropy **23**(12), 1565 (2021)
11. van de Laar, T., Koudahl, M., van Erp, B., de Vries, B.: Active inference and epistemic value in graphical models. Frontiers in Robotics and AI **9**, 794464 (2022)
12. Luttinen, J.: Fast variational Bayesian linear state-space model. In: European Conference on Machine Learning. pp. 305–320. Springer (2013)
13. Millidge, B., Tschantz, A., Buckley, C.L.: Whence the expected free energy? Neural Computation **33**(2), 447–482 (2021)
14. Särkkä, S.: Bayesian filtering and smoothing, vol. 3. Cambridge University Press (2013)
15. Sarkka, S., Nummenmaa, A.: Recursive noise adaptive Kalman filtering by variational Bayesian approximations. IEEE Transactions on Automatic Control **54**(3), 596–600 (2009)
16. Schwartenbeck, P., Passecker, J., Hauser, T.U., FitzGerald, T.H., Kronbichler, M., Friston, K.J.: Computational mechanisms of curiosity and goal-directed exploration. eLife **8**, e41703 (2019)
17. Şenöz, İ., van de Laar, T., Bagaev, D., de Vries, B.: Variational message passing and local constraint manipulation in factor graphs. Entropy **23**(7), 807 (2021)
18. Tschantz, A., Seth, A.K., Buckley, C.L.: Learning action-oriented models through active inference. PLOS Computational Biology **16**(4), e1007805 (2020)
19. Zamiri-Jafarian, Y., Plataniotis, K.N.: A Bayesian surprise approach in designing cognitive radar for autonomous driving. Entropy **24**(5), 672 (2022)

Free Energy in a Circumplex Model of Emotion

Candice Pattisapu*, Tim Verbelen[1*], Riddhi J. Pitliya[1,2], Alex B. Kiefer[1,3], and Mahault Albarracin[1,4]

[1] VERSES Research Lab, Los Angeles, California, 90016, USA
[2] Department of Experimental Psychology, University of Oxford, Oxford, UK
[3] Department of Philosophy, Monash University, Melbourne, VIC, Australia
[4] Department of Computer Science, Université du Québec à Montréal, Montréal, Canada

candice.pattisapu@gmail.com

Abstract. Previous active inference accounts of emotion translate fluctuations in free energy to a sense of emotion, sometimes focusing exclusively on valence. However, in affective science, emotions are often represented as multidimensional. In this paper, we adapt a Circumplex Model of emotion to the Active Inference framework by demonstrating a mapping of free energy into valence and arousal, relating valence to utility less expected utility and arousal to the entropy of posterior beliefs. Under this formulation, we simulate artificial agents engaged in a search task and assign emotional states to them. We show that experimental manipulation of priors and object presence results in commonsense variability in these assignments.

Keywords: active inference · emotional inference · circumplex model of emotion

1 Introduction

Emotions are internal states that influence behavior and cognition [1]. A comprehensive account of intelligence requires a model of emotion, as emotion is intertwined with many higher-level cognitive processes. The aim of this paper is to formalize a model of emotion as a function of free energy. We begin by briefly reviewing foundational accounts of emotion in psychological and active inference literature.

1.1 Psychological Accounts of Human Emotion

Categorical and continuous state space accounts polarize psychological scholarship on emotions. Ekman's Basic Emotions model is a prevailing categorical account, characterized by casting emotions as discrete rather than organized along

* These authors contributed equally to this work.

a continuum [2]. He proposed that six emotions—anger, joy, disgust, fear, sadness, and surprise—comprise the core building blocks of more complex emotional states. Neuroimaging studies cited in support of the Basic Emotions theory associate these core building blocks with specific brain structures, correlating fear with activation of the amygdala, for example [3].

In contrast, a prevailing continuous state space approach is the Circumplex Model [4], which situates emotions in a landscape framed by orthogonal dimensions of valence and arousal. Under this account, emotions have fluid boundaries—e.g., when valence remains low and arousal diminishes, anger transitions to displeasure—as opposed to complexity resulting from compositionality. Supporting neuroimaging research associates unique brain structures and networks with the independent dimensions of arousal and valence. For example, activity in the amygdala has been correlated with arousal, and activity in the orbitofrontal cortex has been correlated with valence [3]. Given that arousal involves reactivity to stimuli, and valence involves appraisal of the value of the stimuli relative to a goal, this localization of arousal and valence dovetails intuitively with their functional roles. Whereas arousal is low-level sensation, valence may require some form of cognitive evaluation.

Approaches to characterizing emotions as discrete cite the evolutionary plausibility of their theories [5], as well as cross-cultural evidence supporting the existence of basic emotions [6]. However, there is a lack of consensus about which basic emotion categories exist—a debate that may be attributed to a lack of agreement on criteria about what a discrete emotion is in the first place [7]. Indeed, discrete models have a difficult time explaining edge cases naturally accommodated by a dimensional view, such as fear-based arousal being interpreted as attraction [8]. In addition, categorical theories can be folded into continuous ones by specifying emotions as attractor states in a continuous state space. Such an approach resolves concerns that referring to emotions as discrete when labeling them linguistically undermines the plausibility of the Circumplex Model. Finally, from a pragmatic perspective, the valence axis of dimensional models are the explanandum of most active inference formulations of emotion. For the reasons mentioned above, we embrace dimensional models in what follows.

1.2 Previous Active Inference Formulations of Emotions

In this section, we summarize select Active Inference formulations of emotion. The work we review addresses valence, but not arousal. We propose that this omission presents an obstacle to engaging a large body of psychological literature on emotion. Our formulation aims to address this gap by integrating both valence and arousal into model of emotions under Active Inference.

Joffily and Coricelli [9] cast valence as an indicator of emotional well-being that emerges from the interaction between the first- and second-order time derivatives of free energy. In their framework, emotional valence provides feedback on an agent's learning process. A rapid decrease in free energy, indicating increasingly accurate predictions, is associated with happiness and suggests that the agent should update its models more quickly by increasing the learning rate.

Conversely, negative valence suggests that a slower learning rate is appropriate. This adaptive mechanism regulates evidence accumulation to optimize learning. In one-armed bandit task simulations, there was a rapid increase and subsequent decline in negative emotional states in the presence of volatility. Agents that used emotional valence to adjust their learning rates better estimated statistical regularities in more volatile environments. The authors conclude that this indicates performance is enhanced by leveraging positive and negative valence, evidencing the adaptive role of emotional experiences. Notably, while their model differentiates some emotion categories, their simulation work leverages the utility of only the positive or negative valence of those emotions.

Hesp et al. [10] proposed a hierarchical Active Inference model of emotional valence. In their approach, valence is defined in terms of changes in the expected precision of the action model, which they term "affective charge". Here, precision refers to confidence in the agent's predictions and actions, which may be regarded as an internal estimate of model fitness. Lower-level state factors, such as sensory inputs, are used to inform higher-level valence representations—"beliefs about beliefs"—which in turn influence the precision of potential action policies. Simulation studies using a T-maze paradigm showed that positive valence led to riskier behavior, interpreted as increased confidence in action. When the reward location was changed, resulting in negative valence, the agent displayed more conservative exploration, indicating a shift in the confidence and model adjustment. Again, the effect of emotions is cast exclusively in terms of positive and negative valence, which compromises the generalizability of the findings to more naturalistic settings.

It is worth mentioning that Smith et al. [11] also explored emotional state inference under Active Inference. In this work, a combination of exteroceptive (external sensory), proprioceptive (body position), and interoceptive (internal sensory) observations are used to infer emotional states, given a hierarchical model in which valence and arousal are presupposed as part of the lower-level observation space. The focus in this work is on the learning of explicit, consciously accessible discrete representations of emotional states, based on explicit feedback supplied by a teacher/experimenter, rather than on the factors that constitute emotional states themselves and the ways in which these states modulate behavior. Therefore, this work does not directly inform our approach.

As noted, the models proposed by Joffily and Coricelli [9] and Hesp et al. [10] focus primarily on valence. We are inspired by their accounts and propose an approach that incorporates arousal in order to capture emotional inference as it is conceived within Circumplex Models. We proceed as follows: In Section 2, we motivate mappings of valence and arousal to specific terms within the free energy functionals of Active Inference, and we demonstrate a transformation of the resulting valence and arousal dimensions of emotion into the space proposed by the Circumplex Model. In Section 3, we describe the setup of our simulation study, in which an artificial agent is tasked with finding an object in various scenarios. In Section 4, we present the results of our simulations, and in Section

5, we discuss the implications of these findings for understanding and formalizing emotional inference and behavior.

2 From Free Energy to a Circumplex Model of Emotion

We hypothesize that the two dimensions of the Circumplex Model can be derived intuitively from an Active Inference agent's free energy levels.

2.1 Active Inference

Active Inference casts perception and action as Bayesian inference [12], where an agent entertains a generative model of its environment and perceives and acts in order to minimize free energy, as defined below, with respect to this model. In general, such an agent's generative model can be written as the joint probability distribution over states s and observations o. However, finding the true hidden cause s given some observation o is typically intractable, as this would require the agent to exhaustively consider all possible worlds and scenarios therein. To address this, one can introduce a simpler but approximate posterior distribution $Q(s|o)$, and minimize the free energy to yield a good approximation:

$$\begin{aligned}\min_{Q(s|o)} F &= \underbrace{D_{KL}[Q(s|o)||P(s|o)]}_{\text{posterior approximation}} - \underbrace{\log P(o)}_{\text{log evidence}} \\ &= \underbrace{-\mathbb{E}_{Q(s|o)}[\log P(o,s)]}_{\text{energy}} - \underbrace{H[Q(s|o)]}_{\text{entropy}} \\ &= \underbrace{D_{KL}[Q(s|o)||P(s)]}_{\text{complexity}} - \underbrace{\mathbb{E}_{Q(s|o)}[\log P(o|s)]}_{\text{accuracy}}\end{aligned} \tag{1}$$

Here P and Q denote a true versus an approximate probability distribution respectively. Effectively, by minimizing free energy, an agent aims to find the model that maximizes accuracy while minimizing the KL divergence between the approximate posterior and prior distributions, i.e. minimizing model complexity. One can rewrite this as optimizing an energy-based model while maintaining the maximum entropy solution for the approximate posterior. This can also be read as maximizing a lower bound on the model log evidence, the bound given by the KL divergence between the approximate $Q(s|o)$ and true posterior distribution $P(s|o)$.

To interact with the environment, an agent also needs to select a sequence of actions or policy π to execute. In Active Inference, agents select policies that minimize expected free energy G:

$$\begin{aligned}&P(\pi) = \sigma(-G(\pi)), \text{ with} \\ &G(\pi) = \sum_{\tau=t+1}^{T} \underbrace{\mathbb{E}_{Q(o_\tau|\pi)}\big[D_{KL}[Q(s_\tau|o_\tau,\pi)||Q(s_\tau|\pi)]\big]}_{\text{(negative) information gain}} - \underbrace{\mathbb{E}_{Q(o_\tau|\pi)}\big[\log P(o_\tau|C)\big]}_{\text{expected utility}}\end{aligned} \tag{2}$$

Here, σ denotes the softmax function, and the expected free energy balances information gain with a prior preference distribution over future outcomes or utility, encoded in C.

2.2 From Free Energy to Valence

We depart from the approach of Joffily and Coricelli [9], who collapse features of valence and arousal when differentiating emotional states by exclusively computing valence from variational free energy and its time derivatives. In addition, we propose a non-hierarchical approach to valence which, unlike the account in Hesp et al. [10], does not invoke policy selection. Our proposal derives a psychologically interpretable description of valence as the difference between the utility of observations given preferred ones and the prior expected utility:

$$Valence\ (V) = \underbrace{\log P(o_t|C)}_{\text{utility}} - \underbrace{\mathbb{E}_{Q(o_t|s_{t-1},\pi)}\big[\log P(o_t|C)\big]}_{\text{expected utility}}, \tag{3}$$

with utility of an observed observation o_t at time step t, an expected utility yielded by the expectation at state s_{t-1}.

This formulation conceptualizes valence as the simple positive and negative experiences humans have when they do and do not encounter what they prefer to observe. The consequence is that positive valence is associated with a "better than expected" outcome. Such a specification of positive valence corresponds straightforwardly to the role of dopamine signaling in the brain, which is known to encode reward prediction errors [13].

2.3 From Free Energy to Arousal

Empirical evidence links arousal to uncertainty vis-a-vis "increases in amygdala activity [14], which can be considered a learning signal [15]" (cited in Feldman-Barrett [16]). Allostasis refers to an internal state of an organism which optimally predicts incoming messages in an effort to maintain homeostasis in a dynamic environment. To achieve a homeostatic state, an active agent needs to move towards an allostatic one, which amounts to updating model parameters to reduce the uncertainty of its posterior beliefs [17]. The empirical relationship between arousal, the amygdala, uncertainty, and the imperative to update model parameters to achieve allostasis motivates casting the entropy H of a posterior distribution as an index of arousal in a dimensional model of emotion. More formally, arousal (A) becomes:

$$Arousal\ (A) = \mathbb{E}_{Q(s|o)}\big[-\log Q(s|o)\big] = H[Q(s|o)] \tag{4}$$

Our interpretation of H as arousal entails that posterior uncertainty is not valenced. This is in line with Feldman-Barrett's claim that arousal is not inherently 'emotional', but rather a signal of uncertainty [16]. Our interpretation also aligns with the lack of simple equivalence between 'emotional' and 'valenced' in Circumplex models.

2.4 Transformation to an Active Inference Emotional State Space

To ground valence and arousal in a Circumplex Model, we transform the Cartesian coordinates (V, A) into the following polar coordinates.

$$r = radius = \sqrt{V^2 + A^2}$$
$$\theta = angle = tan^{-1}(\frac{A}{V})$$

After this transformation, we can represent agents' emotional states in a circle, spacing emotions by both degree and distance from origin, i.e., distance from neutral. The horizontal axis represents the valence dimension, ranging from sad (negative valence) on the left to happy (positive valence) on the right, while the vertical axis represents degree of arousal, ranging from alert (high arousal/uncertainty) to calm (low arousal/certainty). The co-occurrence of valence and arousal results in the varying degrees on the circle with different emotion labels [18]. The cited emotion labels are standardly associated with these orientations on Circumplex Models. Unique to our model, however, the distance from origin represents the "intensity" of the emotion, understood as the strength of the affective response.

Figure 1 shows an example plot of the Circumplex Model in question. The blue trajectory describes the search sequence of a simulated agent who wants to find an arbitrary object and has a precise and correct prior on where to find it. The resulting behavior is an agent that finds the shortest path to the expected

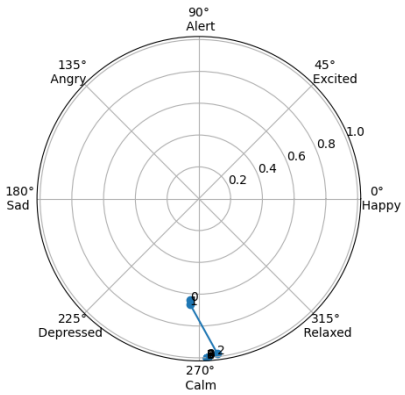
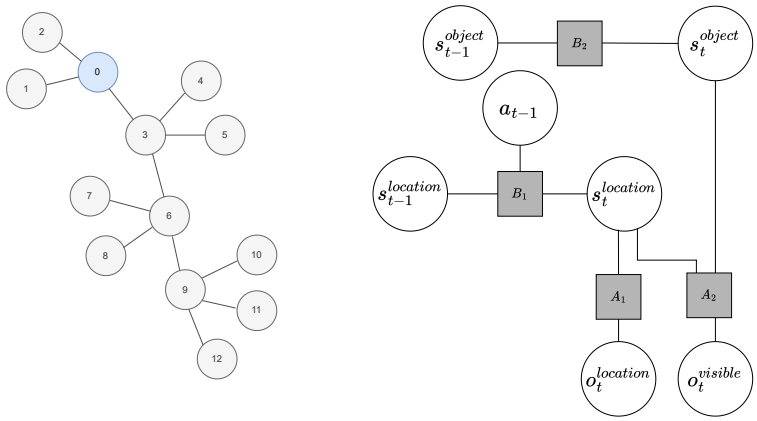


Fig. 1: **Free Energy Transformed Circumplex Model.** In this canonical visualization of a Circumplex Model, degree measure on the circle maps to different emotional states. Here we have illustrated the trajectory of an agent that remains in a "calm" state given arousal and valence values under the free energy formulation introduced above. As implied by the trajectory, a novel contribution of our account is that arousal and intensity are not confounded: Distance to origin reflects the intensity of the emotional state.



(a) (b)

Fig. 2: **Illustration of the graph environment and the agent's factor graph**. (a) Agents are located on a connected graph of locations and need to find an object that might be present at one of the other locations. (b) A factor graph represents the agent's generative model. Two latent state factors that model the agent's location and the object's location, respectively, give rise to two sensory modalities through the likelihood factors of the agent's location (A_1) and whether the object is visible (A_2). The agent's location can change conditioned on move actions (B_1), but the object is static in our experiments ($B_2 = I$).

object location and promptly locates the object at time step 2. The associated state is "calm" for the entire trajectory. In the next section, we will describe the simulated model in more detail, and we will illustrate more diverse scenarios with differing resultant emotional states.

3 Search Agent Simulation

Imagine yourself having lost your wallet. Likely, you will immediately start searching for it. Given your memory and habitual behavior, you may or may not have a good idea about where to find it, and you will experience a range of emotional states as you progress through the corresponding search process. Variations in this scenario are simulated in the following experiments.

3.1 Generative Model

We equip our agent with the generative model represented by the factor graph in Figure 2. The agent has two state factors, one for tracking its own location, and

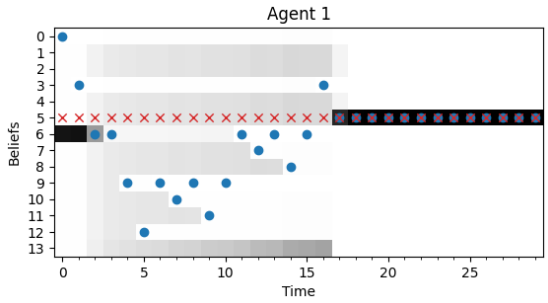


Fig. 3: **Simulation of Scenario 3**. Grayscale shows the agent's belief about the object's location, whereas red x's plot the ground truth object location. The agent's own location is marked with a blue dot. In this case, the agent first has incorrect precise prior beliefs on the object location, then they do not see the object there, and they start searching other locations until it is found.

one for maintaining a belief on the object's location. We model the environment as a connected graph of locations the agent can visit (Fig. 2a). At each time step, the agent can select a location to visit, thought movements are constrained to connected nodes. These transition dynamics are encoded in B_1 (see Figure 2.b). The agent assumes the object is static and therefore the object's transition tensor is the identity matrix, i.e. $B_2 = I$. The agent has two observation modalities. It can observe its current location $A_1 = I$ and whether the object is visible here with a probability p, i.e. $A_2^{oij} = p$ if $i = j$ and $o = visible$, with element o, i, j in A_2 indicating "the probability of outcome o when agent is at location i and object is at location j". In our experiments $p = 0.95$, so the agent is likely to observe the object when it is present.

Scenario	**Object Presence**	**Location Prior**
1	Present	Uniform
2	Present	Correct
3	Present	Incorrect
4	Absent	Maybe Here
5	Absent	Definitely Here

Table 1: **Overview of search agent scenarios given levels of object presence and agent location priors.** Scenario 1: Agent has uniform prior beliefs over object location and object is present in the simulation. Scenario 2: Agent has a correct prior belief over object location. Scenario 3: Agent has an incorrect prior belief on object location. Scenario 4: Agent has a state dim "object not here" and object is not present. Scenario 5: Agent has a state "object must be somewhere" and object is not present.

3.2 Design

The agent always starts at location 0 and has a preference for the outcome "object visible". We simulate 5 different scenarios, varying whether the object is actually present in the environment, as well as the agent's priors. These are summarized in Table 1. We set the agent's prior beliefs on the initial state about the object's location, which can either be uniform (i.e., no idea where the object is) or precise (i.e., recalling where the object is). In the case of a precise prior, this can be correct (i.e., the object is actually there) or incorrect (i.e., the object is either somewhere else or absent). We can also equip the agent's generative model with an additional 'object location' state dimension, which represents the object being absent by mapping to an invisible outcome at all locations. This can be interpreted as providing a prior belief that the object might not be present, versus that the object must be present somewhere.

Figure 3 shows the agent's actions and beliefs over time for a given Scenario 3. The belief about the object's presence at each location is plotted in grayscale, where black represents a belief of 1—high certainty that the object is there—and white represents a belief of 0—high certainty that the object is absent.

4 Results

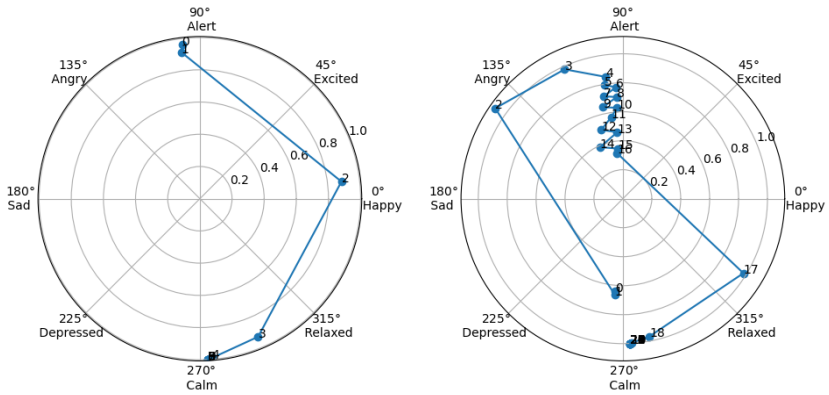
For each of the 5 simulation scenarios, we derived the valence and arousal values from the agent's free energy, and logged the associated emotions. Table 2 summarizes the trajectory of the agent in each scenario through emotional state space. We found that agents' free energy values generally corresponded to a state of alert while searching for the object and happy when it was found. Irrespective of whether the object was located, agents in all scenarios ended in a state of lower arousal and more neutral valence than when they began.

4.1 Object Location Priors and Emotional State

In Scenarios 1-3, the object is present and the agent ultimately locates it. In these Scenarios, we find that the agent begins from a calm state only when it has precise object location priors, irrespective of whether those priors ultimately prove correct. In other words, precise priors are beliefs held with high confidence,

Scenario	**Trajectory Narrative**
1	Alert => Calm
2	Calm => Calmer
3	Calm => Angry => Decreasingly alert => Relaxed => Calm
4	Alert => Calm
5	Alert => Anger <=> Depression

Table 2: **Agent trajectory through emotional state space per scenario.**



(a) Scenario 1 (b) Scenario 3

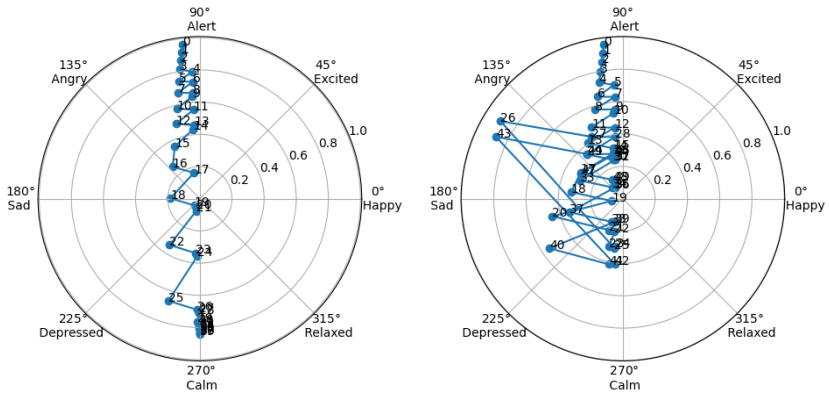
Fig. 4: **Impact of Priors on Emotional State.** In Scenario 1 on the left, the agent has uniform priors and begins alert. In Scenario 3 on the right, the agent begins somewhere between a calm and neutral state, immediately becoming angry upon not finding the object at the location given by their prior.

resulting in a state of ease with low arousal and neutral valence in the absence of countervailing evidence. Moreover, when the precise priors are correct (i.e., Scenario 2, shown in Figure 1), the agent is calm during the entire trajectory, which makes intuitive sense as it predicts that it will find the object soon from the start.

States that are markedly valenced and aroused are triggered only subsequent to prior assumptions being violated, which is possible when priors are either uniform or incorrect (see Fig. 4). For instance, in Scenario 3, the violation of precise priors generated free energy signals that assign the agent to the highly negatively valenced and positively aroused emotional state of anger. When the object is subsequently located, the agent's free energy values correspond to a nearly 180-degree "mood change": Now they are relaxed with high positive valence and low arousal. In Scenario 1, in which the priors are uniform (i.e., Fig., 4a), the agent begins in an alert state. Interestingly, agents locating the object after beginning with uniform priors were not as relaxed when they found it as those who began with precise priors.

4.2 Missing objects and Emotional State

In Scenarios 4 and 5, we simulate a search agent who will never find the object as it is not present in the environment. We conducted these simulations to see whether and how the agent conducts emotional regulation. In Scenario 4, the agent's state factor has a 'not here' bin, in addition to a dimension for every location. Invoking this factor means that the agent expects not to see the object at any of the locations it can visit. In Scenario 5, the agent does not have this



(a) Scenario 4 (b) Scenario 5

Fig. 5: **Impact of object presence on Emotional State.** In Scenario 4 on the left, the agent begins in the alert state, with "maybe here" priors. Similarly, in Scenario 5 on the right, the agent begins alert, with "definitely here" priors. However, in Scenario 4, the agent is ultimately assigned to a state of calm, which can be interpreted as accepting that there is no object to find. In contrast, the agent in Scenario 5 cycles between anger and depression when searching all locations and still not finding the object.

dimension, hence it has a structural prior that the object should always be at one of the locations. This has some interesting consequences for the emotional states the agent visits, as illustrated in the contrast between the Circumplex plots in Figure 5.

While both agents begin in the alert state, the agent with "maybe here" priors (Scenario 4) spends less time in strongly negatively valenced states and ends in a state of much lower arousal upon failing to find the object. A psychological interpretation of this result is that resignation grows as evidence that the agent will not find the object increases, given that the agent knows there is a possibility that the object is simply not around.

By contrast, the agent with "definitely here" priors (Scenario 5) spends more time in strongly negatively valenced states and ends in a state of strong arousal upon failing to find the object. Interestingly, both anger and depression in this agent spike at later time steps (26 out of 43 and 20 out of 40, respectively), indicating that there is a pull towards neutral arousal resisted when the agent is convinced that the object must be somewhere. While this agent becomes transiently more calm as its search eliminates possibilities, providing false hope of finding the object, it ultimately cannot accept an "object not present" outcome, and it is in a more highly aroused state, i.e., less at peace, when the simulation terminates. Critically, all emotional states are not being interpreted as "experienced" by the agent, but rather assigned on the basis of intuitive formulations of free energy in keeping with external literature on the psychology of emotions.

5 Discussion

In this paper, we proposed the existence of a Circumplex Model of emotion formulated within the Active Inference framework by casting both valence and arousal within the parameters of free energy. Simulation studies provided evidence that a model which includes arousal facilitates a commonsense ascription of emotion trajectories to agents performing basic search tasks.

Considering these scenarios in more detail, uniform and incorrect priors (Scenarios 1 and 3, respectively) result in agents entering equivalent positively valenced states upon finding the object. However, the agent in Scenario 3 finishes considerably less aroused. Indeed, the emotion trajectory of the agent in Scenario 3 involves a mood swing from the strongly negatively valenced state of anger to a region of neutral valence. Because this transition cannot be described in terms of valence alone, this result underscores the importance of representing arousal in Active Inference accounts of emotion. In addition, the agent in Scenario 3 becomes successively less aroused before finding the object. A description of emotional states exclusively in terms of valence would not capture the fact that, after having a prior expectation violated, an agent may return to an emotional state of being alert, a state which dwindles to resignation as the search task progresses.

This work faces a couple of key limitations. One is that the scenarios are simple and the range of emotion evidenced is correspondingly narrow. Another is that our findings are not benchmarked against human data. Instead, we rely on face validity to make sense of the attribution of emotional states to simulated agents. Finally, our current model lacks an account of metacognitive awareness, from which valence more naturally derives. We will address these limitations in future work.

Though it faces these and other limitations, the account we've presented offers a unique contribution to the literature. Some research suggests that arousal and intensity are distinct [19], and consistent with this work, our model straightforwardly disambiguates the two by positing intensity as distance from the origin of neutrality—a third characterization of an emotional state. An account of emotional intensity may have implications for other cognitive processes, such as event segmentation in an anthropomorphic account of episodic memory in simulated agents, as more intense emotional experiences are more likely to be encoded and recalled [20].

In future work, we aim to measure how emotion impacts inference and learning. In addition, we'd like to explore temporal dimensions of emotion and emotional state attribution, e.g., how one state tends to transition to another. Finally, future work may explore emotions among simulated agents in social contexts. For example, finely-grained inference of an agent's own emotions is a prerequisite for empathy. Because our account leverages the arousal/valence distinction to define a granular emotional state space, it may serve as a foundation for simulating empathy in multi-agent scenarios.

References

1. J. Martínez-Miranda and A. Aldea, "Emotions in human and artificial intelligence," *Computers in Human Behavior*, vol. 21, p. 323–341, Mar. 2005.
2. P. Ekman and D. Cordaro, "What is meant by calling emotions basic," *Emotion Review*, vol. 3, p. 364–370, Sept. 2011.
3. K. A. Lindquist, T. D. Wager, H. Kober, E. Bliss-Moreau, and L. F. Barrett, "The brain basis of emotion: A meta-analytic review," *Behavioral and Brain Sciences*, vol. 35, p. 121–143, May 2012.
4. J. A. Russell, "A circumplex model of affect.," *Journal of Personality and Social Psychology*, vol. 39, p. 1161–1178, Dec. 1980.
5. R. Plutchik, *Emotion, a Psychoevolutionary Synthesis*. Harper & Row, 1980.
6. J. A. Russell, "Culture and the categorization of emotions.," *Psychological Bulletin*, vol. 110, no. 3, p. 426–450, 1991.
7. A. Ortony, "Are all "basic emotions" emotions? a problem for the (basic) emotions construct," *Perspectives on Psychological Science*, vol. 17, p. 41–61, July 2021.
8. D. Dutton and A. Aron, "Some evidence for heightened sexual attraction under conditions of high anxiety," *Journal of personality and social psychology*, vol. 30, pp. 510–7, 10 1974.
9. M. Joffily and G. Coricelli, "Emotional valence and the free-energy principle," *PLOS Computational Biology*, vol. 9, pp. 1–14, 06 2013.
10. C. Hesp, R. Smith, T. Parr, M. Allen, K. J. Friston, and M. J. D. Ramstead, "Deeply felt affect: The emergence of valence in deep active inference," *Neural Computation*, vol. 33, p. 398–446, Feb. 2021.
11. R. Smith, T. Parr, and K. J. Friston, "Simulating emotions: An active inference model of emotional state inference and emotion concept learning," *Frontiers in Psychology*, vol. 10, Dec. 2019.
12. T. Parr, G. Pezzulo, and K. J. Friston, *Active Inference: The Free Energy Principle in Mind, Brain, and Behavior*. The MIT Press, 03 2022.
13. W. Schultz, "Dopamine reward prediction error coding," *Dialogues in Clinical Neuroscience*, vol. 18, p. 23–32, Mar. 2016.
14. C. D. Wilson-Mendenhall, L. F. Barrett, and L. W. Barsalou, "Neural evidence that human emotions share core affective properties," *Psychological Science*, vol. 24, p. 947–956, Apr. 2013.
15. S. S. Y. Li and G. P. McNally, "The conditions that promote fear learning: Prediction error and pavlovian fear conditioning," *Neurobiology of Learning and Memory*, vol. 108, p. 14–21, Feb. 2014.
16. L. F. Barrett, "The theory of constructed emotion: an active inference account of interoception and categorization," *Social Cognitive and Affective Neuroscience*, p. nsw154, Oct. 2016.
17. A. W. Corcoran, G. Pezzulo, and J. Hohwy, "From allostatic agents to counterfactual cognisers: active inference, biological regulation, and the origins of cognition," *Biology & Philosophy*, vol. 35, Apr. 2020.
18. J. Posner, J. A. Russell, and B. S. Peterson, "The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology," *Development and Psychopathology*, vol. 17, Sept. 2005.
19. G. Raz, A. Touroutoglou, C. Wilson-Mendenhall, G. Gilam, T. Lin, T. Gonen, Y. Jacob, S. Atzil, R. Admon, M. Bleich-Cohen, A. Maron-Katz, T. Hendler, and L. F. Barrett, "Functional connectivity dynamics during film viewing reveal common networks for different emotional experiences," *Cognitive, Affective, & Behavioral Neuroscience*, vol. 16, p. 709–723, May 2016.

20. K. KaYan, H. Ginting, and C. Cakrangadinata, "It is fear, not disgust, that enhances memory: Experimental study on students in bandung," *ANIMA Indonesian Psychological Journal*, vol. 31, p. 77–83, Jan. 2016.

Epistemic Value Anticipation into the Deep Active Inference Model

Nikita Fedosov[1,2][0000−0002−2456−9651], Alexey Voskoboynikov[1], and Alexey Ossadtchi[1,2][0000−0001−8827−9429]

[1] Center for Bioelectric Interfaces, Higher School of Economics, Moscow, Russia
[2] AIRI, Artificial Intelligence Research Institute, Moscow, Russia

Abstract. This article introduces a novel mathematical and computational framework for epistemic value calculation within deep active inference models. We focus on a visual foraging problem in a static environment, using toy and real-world MNIST datasets to explore the drawbacks and advantages of the standard method compared to our proposed approach. Testing with relevant metrics, our approach demonstrates improved results in the considered scenarios.

Keywords: Deep Active Inference · Information Gain · Epistemic Value · Visual Foraging.

1 Introduction

1.1 Free Energy Principle

This article explores a method for calculating information gain in deep active inference (DAI) model, which employs neural networks to approximate the probability densities of an agent performing Bayesian inference. These has proven effectiveness for learning environment structures from scratch but also presents challenges, such as efficient neural network composition, action generation, and replication of the epistemic behavior [2].

The active inference (AIF) is the corollary of the Free Energy Principle (FEP) and states that any self-organizing system that is at equilibrium with its environment must minimize its free energy [3]. Mathematically FE is the upper bound of the negative log evidence of observations $\ln p(o)$, which is the entropy of observations under the assumption of ergodicity (existence in a long period of time)[6], so the FEP and AIF can be interpreted as laws for a natural resistance to the disorder.

In the context of agent-environment interaction, FEP utilizes the Markov blanket concept [7] separating the agent's inner states s_t from the outer states and complex dynamics of the world through a statistical boundary. This boundary is formed by the observations (sensations) o_t and active states a_t. The agent, immersed in a stochastic environment with such limited interaction channels, must maximize the information exchange with the outer world and reduce the

uncertainty to adapt to a constantly changing and non-deterministic environment, thereby introducing the concept of information gain.

Given the defined states and actions, the FE for the agent at time step t is [11]:

$$F_t = D_{KL}[q(s_t, a_t)||p(s_t, o_t, a_t)] = D_{KL}[q(s_t)||p(s_t)] + \\ + \mathbb{E}_{q(s_t)} D_{KL}[q(a_t|s_t)||p(a_t|s_t)] - \mathbb{E}_{q(s_t)} \ln p(o_t|s_t) \geq -\ln p(o_t) \quad (1)$$

This equation enables training the model for perception by minimizing FE as an upper bound of the negative log-evidence of observations.

To enable future planning the expected free energy (EFE) has been suggested [9]:

$$G_\tau = \mathbb{E}_{q(o_\tau, s_\tau)}[\ln q(s_\tau) - \ln p(o_\tau, s_\tau)] \quad (2)$$

Where $\tau = t + 1 : T$ is an unobservable future time index. This equation (2) has the following useful decomposition:

$$G_\tau = -\mathbb{E}_{q(o_\tau, s_{t+1:\tau-1})}[\underbrace{\ln p(o_\tau)}_{\text{Extrinsic value}} + \underbrace{D_{KL}[q(s_\tau|o_\tau, s_{0:\tau-1})||q(s_\tau|s_{0:\tau-1})]}_{\text{Epistemic value}}] \quad (3)$$

Where we change the generative true posterior distribution $p(s_\tau|o_\tau)$ to the approximate posterior $q(s_\tau)$ as we have them in our generative model and vise versa. This decomposition separates expected reward (extrinsic value) and intrinsic motivation (epistemic value). The intrinsic motivation encourages the agent to interact with the environment to resolve uncertainty about underlying states and gives the motivation to act in a sparse-reward environment. This is an essential feature of AIF distinguishing it from most other reinforcement learning approaches, endows the agent with different forms of natural curiosity [4, 5, 8].

1.2 Information Gain in active inference model

Scaling the AIF framework to complex environments remains challenging. Previous works [1, 2, 12] tried to overcome the problem and introduced deep active inference models (DAI) and approximated the unknown probability distributions with normal distributions. Despite the successful demonstration the main aim of the papers, i.e the IG calculation, was not sufficiently covered. The source of the problem is that while in the classic discrete probabilities AIF framework the calculation of epistemic value is straightforward, in the DAI case due to the ancestral sampling scheme employed during training the estimation of the epistemic value is much more challenging. In [1] the authors used the entropy of sensory states whose distribution was approximated as normal which is likely not the case in reality. In [2] the IG was calculated as the divergence between the prior and posterior from (3). This way is more formal and can be easily

interpreted as the precision of the measurement based on the observation, we call this method "Information Gain (Instant)", IGI.

The drawback of this method is that it has the important difference with the IG term which can be calculated into the discrete AIF model, where the states on the initialization encode the probability on the environment states as a whole. Into the DAI the state distribution encodes the local information needed to reconstruct the current observation so the IGI tells only on the precision of the currently observed values.

We reframe the IGI method for the IG estimation and by endowing it with prospective exploration which considers the most likely visited parts of the environment and therefore can potentially better find the clues to resolve uncertainty of the world as a whole. We name the method "Epistemic Value (Anticipatory)", EVA.

We focus on a simplified environments which have no inner dynamics and the next observation given the action does not depend on the previous observation, this allows as to significantly simplify the computational schemes for both IGI and EVA. We demonstrate the effectiveness of EVA comparing to the IGI qualitatively on the example of a toy environment where four stove plates are connected to the power line with a pattern which makes it possible to measure the temperature of all the plates within minimal number of steps. Then we explore the two methods using more complex MNIST handwritten digits dataset. Here we use the natural metric provided by the DAI model for the quantitative analysis, which shows the reliability of the EVA in the visual search task (static environment) and superiority to both random and the IGI-based policies. Finally, we discuss the results, limitations of both methods, and future perspectives.

2 Methods

2.1 Theoretical derivation of anticipatory epistemic value

We begin with the FE model used in the work formalization. We explicitly write the dependency on the state history untill the timestep t, and the FE equation (1) can be rewritten:

$$F_{t,0} = D_{KL}[q(s_t)||p(s_t|s_{0:t-1}, a_{0:t-1})] + \mathbb{E}_{q(s_t)} p_\theta(o_t|s_t) \tag{4}$$

Here the second index 0 indicates that the state s_t is predicted from the previous time step and action. We explicitly use the autoregressive model where the state s_t depends on the full history of states and actions, as implicitly introduced in [1] with LSTM network in the transition and posterior model.

This expression was previously used to train the DAI agent, but the key difference in our model is the parallel training of the alternative FE model for predictions $k = 0 : T - t$ steps to the future:

$$F_{t,k} = D_{KL}[q(s_{t+k})||p(s_{t+k}|s_{0:t-1}, a_{0:t-1+k})] - \mathbb{E}_{q(s_{t+k})} \ln\, p_\theta(o_{t+k}|s_{t+k}) \tag{5}$$

We emphasize with θ that the likelihood model is the same for all the k alternative factorizations. Essentially, this is the same model but now the prior and posterior can depend on the history of the states and actions with some lag to the past.

Next, we move to the epistemic term of the EFE (3), where the observation can only be generated by our generative model. We show that the alternative factorization in the generative model changes the IG representation:

$$\begin{aligned} IG_{\tau=t+2:T} = -\mathbb{E}_{q(o_\tau, s_{t+1:\tau-1})}[D_{KL}[q(s_\tau|o_\tau, s_{0:\tau-1})||p(s_\tau|s_{0:t})]+ \\ + \mathbb{E}_{q(o_\tau, s_{t+1:\tau-1})} \sum_{k=1:\tau-1-t} [\ln\, p(s_\tau|s_{0:t+k})]- \\ - \mathbb{E}_{q(o_\tau, s_{t+1:\tau-1})} \sum_{k=1:\tau-1-t} [\ln\, p(s_\tau|s_{0:t+k})]] \end{aligned} \quad (6)$$

We can use the first term from (6) as an alternative to the IG from (3), which should suffer less from the ancestral sampling scheme since the lagged distribution $p(s_\tau|s_{0:t})$ encodes a wider number of state leaves relative to the root at time t. However, we want to separate this term into two parts: one encoding immediate IG and the other encoding prospective IG. When we sum the IG_τ over all future timesteps $t+1:T$, we can rearrange the terms of the expression:

$$\begin{aligned} IG = -\sum_{\tau=t+2:T} [\underbrace{\sum_{k=1:\tau-1-t} \mathbb{E}_{p(s_{t+1:t+k})}[D_{KL}[p(s_\tau|s_{0:t+k})||p(s_\tau|s_{0:t+k-1})]]}_{\text{EVA}}]- \\ - \sum_{\tau=t+1:T} [\underbrace{\mathbb{E}_{q(o_\tau, s_{t+1:\tau-1})}[D_{KL}[q(s_\tau|o_\tau, s_{0:\tau-1})||p(s_\tau|s_{0:\tau-1})]]}_{\text{IGI}}] \end{aligned} \quad (7)$$

After this step, we can distill anticipatory epistemic value (EVA, "anticipatory" means that agent anticipates and compares the future state distributions before and after choosing a certain action given the memory state generating the action), which is crucial for information collection on the environment, from the instant IG (IGI), which only tells about the precision of measurement. In practice, both terms can be used with different relative weights. Now we have EVA for each time step into the future:

$$IG_{\tau=t+2:T} = -\sum_{k=1:\tau-1-t} \mathbb{E}_{p(s_{t+1:t+k})}[D_{KL}[p(s_\tau|s_{0:t+k})||p(s_\tau|s_{0:t+k-1})]] \quad (8)$$

The presented EVA requires complex tree computation as it adds an additional dimension for summarizing compared to the IGI. We do not explore the computational and theoretical limits for the full-mode EVA estimation but

demonstrate the method's fitness in a constrained environment with a key property: the observation's distribution of the environment depends only on the chosen action. This allows us to radically reduce equation (8). There is no need to estimate temporally deep components (for both EVA and IGI) as they minimally depend on the action history; therefore, we can approximate the full EVA by estimating it within two imaginary steps. One can think of it as setting the loss decay factor to 1 for the nearest future and 0 for other steps:

$$IG \approx -\mathbb{E}_{p(s_{t+1})}[D_{KL}[p(s_{t+2}|s_{0:t+1}, a_{0:t+1})||p(s_{t+2}|s_{0:t}, a_{0:t+1})]] \tag{9}$$

One might ask how we obtain action a_{t+1} without having s_{t+1} - In practice, we estimate $p(s_{t+1}|s_t)$ and then apply habitual policy $p(a_{t+1}|s_{0:t+1}, a_{0:t})$ yielding the required action distribution which we then reuse for the EVA computation.

2.2 Environments

We use two environments: a very simple one and a more realistic one. Both are for epistemic foraging task where the agent chooses which area of the image to measure at each step. There is no extrinsic reward; only IG influences the agent's decisions, motivating it to explore the environment as efficiently as possible. Intuitively understandable qualitative results are demonstrated in the toy stove environment, while specific metric is used for the MNIST environment.

Toy environments For didactic purposes and to directly compare two methods of IG estimation without complex metrics, we use a toy environment with 4 stove plates, each with its own temperature (ranging from very cold to very hot) (see Figure 1).

The values of the plates form specific pattern upon initialization. Values of plates 3 and 4 are strongly correlated, plate 2 maintains its own temperature, and plate 1 is broken, changing temperature randomly each time step. During a series of measurements, each plate (except the broken one) maintains its initial temperature and can be observed by the agent one per step.

More specific, the marginal distribution of each stove plate upon initialization is given by the expression: $p(o) = \sigma(-(L + \mu) \cdot \gamma - \mu \cdot \frac{1}{1+\gamma})$ Here, $\sigma(.)$ represents the sigmoid function, L is the genlogistic distribution, and the parameter γ is set to 0.05. The center μ was unique for the second plate and shared by the third and fourth plates. It was drawn from a Gaussian distribution with a standard deviation of 1.4 during initialization. Additionally, the value of the first plate is updated at each time step according to the aforementioned distribution. A small Gaussian noise with the standard deviation of 0.02 was also added to each observation.

The agent's goal is to learn all plate values in the minimal number of steps, with the training limit of 13 steps. An optimal strategy involves probing plate 3 or 4 first (as they provide values for two plates by one measurement), then probing plate 2. This nearly fully resolves the environment's state since plate 1 is fully random and provides no useful information at all. The Results section shows derived policies governed by the two versions of the IG term.

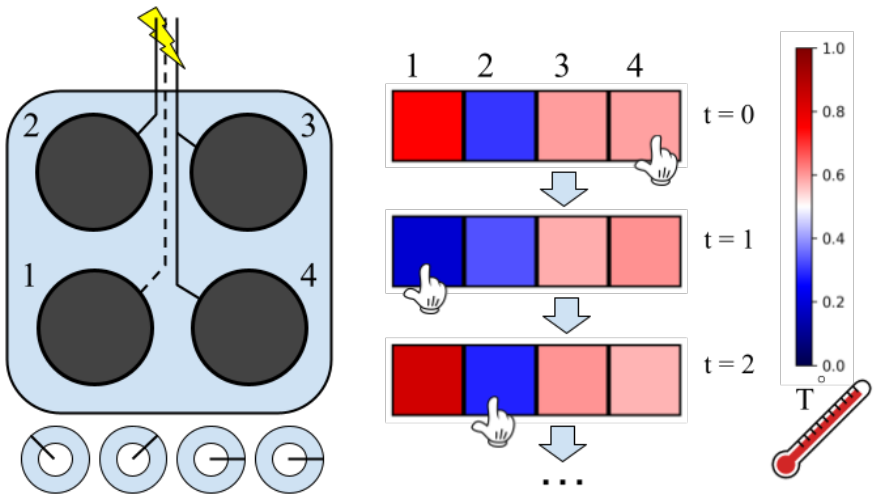


Fig. 1. The toy stove environment illustration. Each plate has its own temperature: one is broken and does not maintain temperature, one is normal, and two are connected, sharing close values. The agent can probe one plate per step

MNIST environment Next, we use a real-world based static dataset, simulating the sequence of decisions an agent makes to explore an image from the MNIST dataset 2. In this environment, the agent chooses an area of the picture to observe at each timestep. The entire image is 24x24 pixels (after cropping 2 uninformative pixels), and each observation covers an 8x8 pixel area, making it impossible to see the whole digit at once (see Figure 2).

We introduce a small amount of Gaussian noise to the agent's actions and make the observation space continuous two-dimensional, as the 24x24 pixel discrete space is challenging to work with. Thus, the agent chooses coordinates within the [0:1] range on both x and y axes. We trained the perception and action models on sequences of length 30 and tested the IG methods on sequences of length 10.

2.3 Model architecture and process of training

We employed two similar models for both the toy and the real-world environments, based on deep neural networks that parameterize probability distributions (prior, posterior, likelihood, habitual action network) with Gaussian distributions. While these models are similar to those from previous works [1, 2], they differ in a few details. Firstly, we explicitly used an autoregressive model within the prior (dynamical) network, endowing the model with memory of the state history. In [1], an LSTM model was implicitly employed for a similar purpose.

Secondly, we did both the posterior (inference) and prior (temporal dynamic) networks one instance, with an additional signal channel indicating the presence

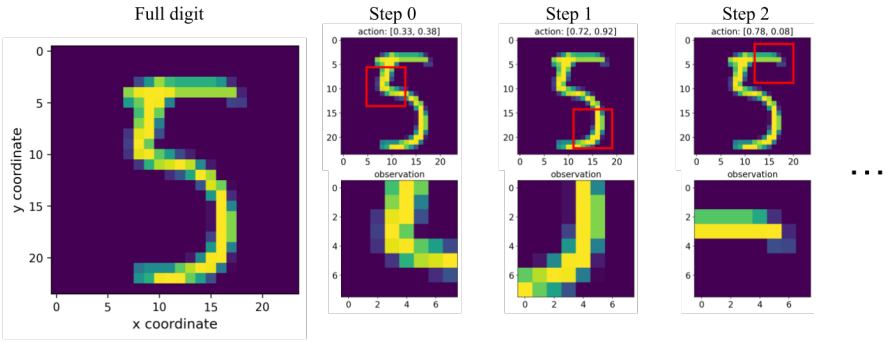


Fig. 2. The MNIST environment illustration. The agent sequentially explores the digit by choosing coordinates for each observation.

or absence of an observation at the input of the network. This simplification reduces the number of parameters and potentially facilitates learning similar patterns within the hidden layers for both the posterior and prior distributions.

The model architecture is presented on figure 3

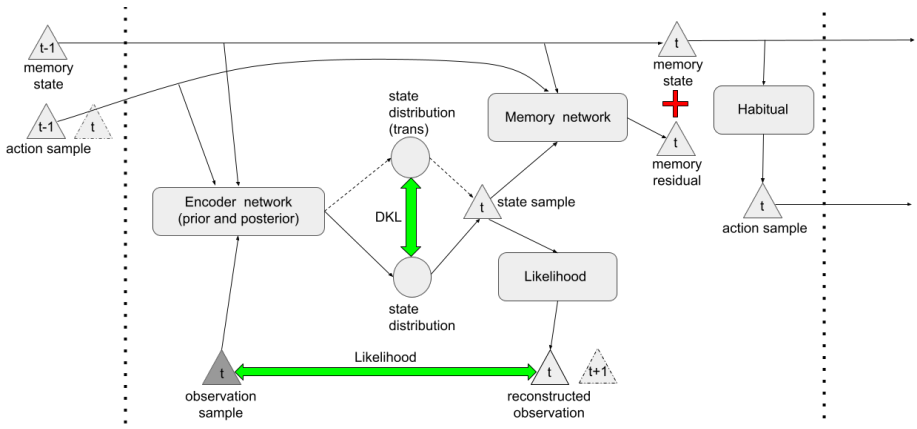


Fig. 3. The schematic representation of the DAI model used. Triangles represent generated samples, circle - Gaussian distributions, green arrows - FE loss (EFE is not represented), rectangles - neural networks. Note in the MNIST environment the auxiliary reconstructor network was used also which is not drawn for clarity.

Our model comprises encoder, memory, likelihood, habitual and reconstructor networks (only for the MNIST environment). Note the residual difference scheme into the memory state update, which implements our potentially infi-

nite order autoregressive model. All other model parameteres and details can be found in the python code on github via the e-link:

$https://github.com/nfedosov/digit\_Xplore\_final$

In the toy environment with a discrete 4-dimensional action space, the action (habitual) network outputs the probabilities of actions given the current state history, trained directly based on the evaluation of EVA for each possible action. This action selection method allows the explorative behaviour on each run of the model. For the MNIST environment, given the continuous action space, the habitual network was trained using the method described in [10], which allows to increase entropy of continuous action distribution in cases like ours (in principle, any other entropy-increasing method potentially can be used). All networks were trained in parallel, minimizing their FE and EFE estimations until all losses stabilized. For the MNIST dataset, we split dataset on the Train (50,000 images) and the Test set.

2.4 IG comparison for the MNIST environment

While the toy environment allows a qualitative matching EVA and IGI models against the optimal measurement strategy, no intuitive strategies exist for the MNIST dataset. Therefore, we introduce a metric to evaluate the IG gained with any model.

IG represents the resolution of uncertainty about the state of the world. A good model should accumulate more information about the world in a shorter period, thereby better predicting future observations. The model has more information on the future if there is less difference in the prior on the future states relative to the posterior. Therefore, we utilize the trained perception model and introduce the Unexpectedness metric (UNEX-metric, U), calculated as the complexity term in the FE formulation (1):

$$U = \int_a D_{KL}[q(s|o,a)||p(s|a)]da \tag{10}$$

The complexity is integrated over all possible actions (picture coordinates), providing a quantitative estimation of the overall unexpectedness of observation for the all the parts of the image. We calculate this metric and compare it for three policies: based on IGI, EVA, and random action selection. The optimal action for each time step was determined by the maximal value of the corresponding IG metric (IGI or EVA) over all possible actions (which differs from the policy during training, where the action was provided by the habitual network at each time step).

Using the U-metric, we conducted the paired Wilcoxon test for each time step on 100 estimations of IGI and EVA from the MNIST test set. For the test, and visualization of U-metric distribtuion, we first divide the IGI and EVA IG values by their mean for the same digit and then subtract the mean, resulting in symmetric values.

3 Results

Both models for the toy and MNIST environments were trained until they achieved stable, low FE and EFE losses. After training, the models exhibited generative behavior, accurately predicting observation patterns based on input actions and observation history (Figure 4). This validates the model architecture and the subsequent IG estimations, which are based on the perception model and depend on the model's ability to memorize the history of states.

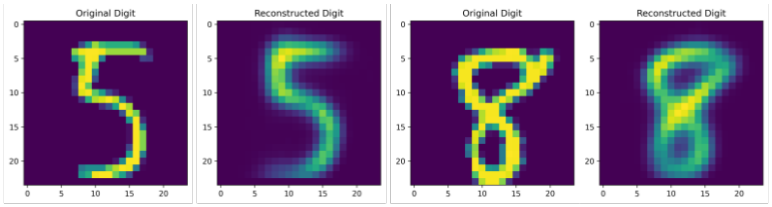


Fig. 4. Two examples of the original digits from the MNIST dataset reconstructed by DAI model after 30 steps with a random policy

3.1 The EVA provides correct optimal policy into the toy stove environment in contrast to IGI

Figure 5 illustrates the difference in IG evaluation between the two methods.

EVA ranks the plate IG values in line with the optimal policy (see methods) and ignores the broken plate, as it carries no information about the other plates or its own value at other times. It assigns moderate value to the second plate, with maximal realtive IG value on the second step after learning the states of the 3rd and 4th plates. Finally, after two steps, it has gained knowledge about all predictable values and does not evaluate any significant IG values over the plates.

In contrast, IGI estimation is ineffective in our case. We expect that on the first step, the expected IG for all plates would be equal, given their equivalent marginal distributions. However, possibly due to model imperfections or other reasons, only the broken plate shows a high IG value relative to the other plates, leading to no exploration if the agent follows the IGI estimation policy.

3.2 Comparizon of two methods on the MNIST dataset

We tested the surprise value on 100 pictures from the test subset of the MNIST dataset, using the first 10 steps for three suboptimal policies provided by EVA, IGI, and uniformly random actions. Figure 6 demonstrates two alternative paths for EVA and IGI.

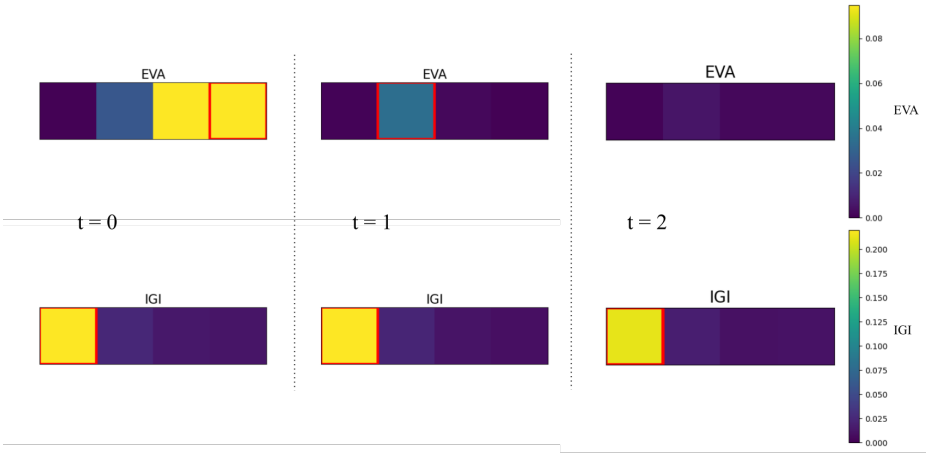


Fig. 5. Illustration of IG estimated by two methods: EVA (upper) and IGI (lower). The red frame indicates the action chosen by the agent based on the maximal IG value.

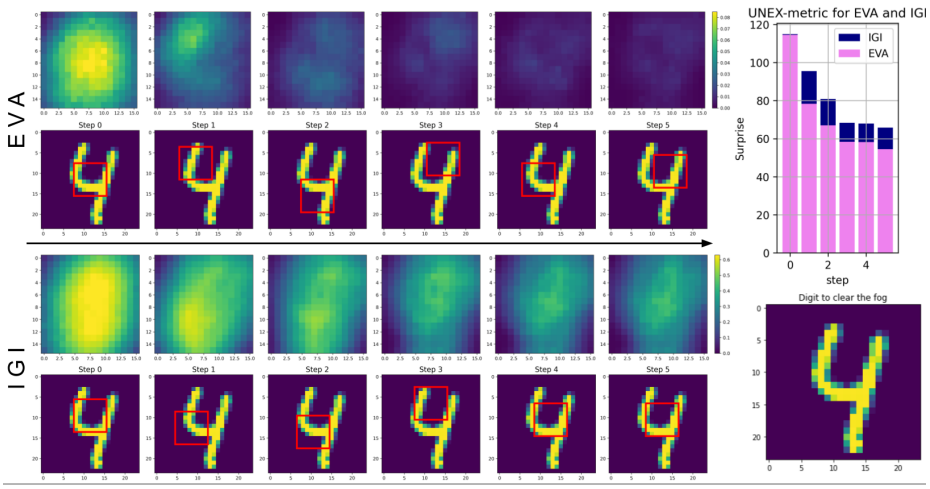


Fig. 6. IG estimated by two methods: EVA (upper) and IGI (lower). The red frame on the digit image indicates the action chosen by the agent based on the maximal IG value. The heatmap above the digits represents the histogram of IG values. The histogram of Surprise for each time-step is in the upper-right corner, and the original digit in large scale is in the lower-right corner

Figure 6 shows the successful performance of EVA on the digit "4", which has common feature with digits like "9" and other. The EVA-based policy directs the agent to first look at the center of the image, then explore the edges (steps 1, 2, 3), effectively observing the full digit in 4 steps. IGI works similarly but pays less attention to the edges (only step 3 explores the upper part of the digit), preventing full exploration, which can be detrimental for the overall results of the method test.

Although EVA often outperforms IGI, there are instances where the opposite is true. To verify the statistical significance of the differences between the methods, we conducted tests on 100 digits. After normalizing the S-loss (see methods), we performed the Wilcoxon paired test for each time step. The results are shown in Figure 7.

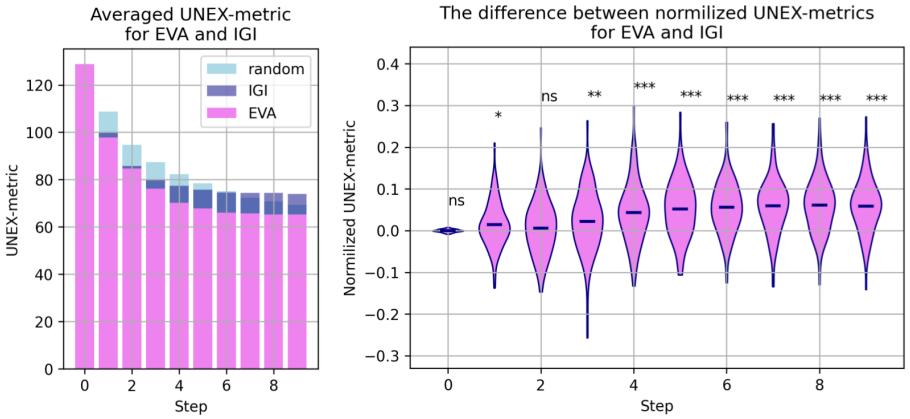


Fig. 7. The histogram (left) of the averaged across set of test digits UNEX-metric for three policies: random, IGI-based, EVA-based. The normilized UNEX-metrics distributions (right) with statistical significance indication

4 Discussion

We conducted a series of demonstrative experiments to show the feasibility of the EVA IG metric in resolving uncertainty in a visual foraging task. The toy environment demonstrated EVA following a simple, intuitively correct strategy, while IGI preferred the non-informative plate at every time step.

We observed similar behavior of the IGI policy in the MNIST environment, characterized by a permanently high attention to high-entropy areas across different samples from the environment. We did not investigate this issue rigorously and cannot definitively say whether the model architecture plays a role, if it's due to the likelihood, or if it's a more fundamental problem with IGI. Therefore,

further comparisons between EVA and IGI should be conducted in simpler or more controlled environments and also using other metrics.

While our metric gave correctly equal values for random, EVA-based, and IGI-based strategies at the first step, and confirmed significant superiority of EVA on most of the subsequent steps, statistical testing with the Surprise metric yielded two unexpected results: the lack of difference in surprise between IGI and EVA on the 2nd step (despite significant difference on the previous one), and high surprise values for the IGI-based policy at the later steps, that appeared to be even higher than for the random policy. We speculate the latter can be explained by the imperfections in the overall models (including the likelihood) required for IGI computation, a drawback from which the EVA method does not suffer. The first problem might be due to EVA's dependence on having useful information to predict the future; on early steps, therefore when forward predictions have very blurred distributions, the IG metric's quality suffers.

We highlight that our evaluation considered only static images (environments), where the agent's actions did not affect the world's state. This allowed us to test the method's principles. Further research is needed to compare the methods in dynamic environments, where the action history affects world states.

In conclusion, our EVA method better utilizes the predictive ability and complex architecture of the DAI agent by using the history of observations as a basis for IG calculation. In summary, while IGI directs the agent to "look at the point with maximal entropy of the predicted observation," EVA directs the agent to "look at the point which can help to reduce the future state probabilities". We hope the feasibility and relevance of the EVA method will be confirmed in more complex real-world AIF tasks and the method will find its niche in the rapidly evolving field of active inference.

References

1. Çatal, O., Wauthier, S., De Boom, C., Verbelen, T., Dhoedt, B.: Learning generative state space models for active inference. Frontiers in Computational Neuroscience **14**, 574372 (2020)
2. Fountas, Z., Sajid, N., Mediano, P., Friston, K.: Deep active inference agents using monte-carlo methods. Advances in neural information processing systems **33**, 11662–11675 (2020)
3. Friston, K.: The free-energy principle: a unified brain theory? Nature reviews neuroscience **11**(2), 127–138 (2010)
4. Friston, K.J., Daunizeau, J., Kiebel, S.J.: Reinforcement learning or active inference? PloS one **4**(7), e6421 (2009)
5. Friston, K.J., Lin, M., Frith, C.D., Pezzulo, G., Hobson, J.A., Ondobaka, S.: Active inference, curiosity and insight. Neural computation **29**(10), 2633–2683 (2017)
6. Karl, F.: A free energy principle for biological systems. Entropy **14**(11), 2100–2121 (2012)
7. Kirchhoff, M., Parr, T., Palacios, E., Friston, K., Kiverstein, J.: The markov blankets of life: autonomy, active inference and the free energy principle. Journal of The royal society interface **15**(138), 20170792 (2018)

8. Mazzaglia, P., Catal, O., Verbelen, T., Dhoedt, B.: Curiosity-driven exploration via latent bayesian surprise. In: Proceedings of the AAAI conference on artificial intelligence. vol. 36, pp. 7752–7760 (2022)
9. Millidge, B., Tschantz, A., Buckley, C.L.: Whence the expected free energy? Neural Computation **33**(2), 447–482 (2021)
10. Nikita, F., Voskoboynikov, A.: Deep active inference agent with continuous action space. In: 2023 IEEE Ural-Siberian Conference on Computational Technologies in Cognitive Science, Genomics and Biomedicine (CSGB). pp. 215–220. IEEE (2023)
11. Smith, R., Friston, K.J., Whyte, C.J.: A step-by-step tutorial on active inference and its application to empirical data. Journal of mathematical psychology **107**, 102632 (2022)
12. Tschantz, A., Baltieri, M., Seth, A.K., Buckley, C.L.: Scaling active inference. In: 2020 international joint conference on neural networks (ijcnn). pp. 1–8. IEEE (2020)

Contextuality, Cognitive engagement, and Active Inference

Avel Guénin–Carlut[1,2,3][0000−0001−8239−7264]

[1] Department of Engineering and Informatics, University of Sussex
[2] Kairos Research
[3] Active Inference Institute
avel@kairos-research.org
https://avelguenin.github.io/

Abstract. We revisit existing criticism of the Free Energy Principle and Active Inference through the concept of cognitive contextuality. Contextuality describe the property of physical states which are brought about by the very act of their observation, such as the position or momentum of individual particles in quantum settings. Based on conceptual and physical argument, we propose that contextuality drives the construction of cognitive semiotics. As such, it constitutes a fundamental component of cognition which any formal theory thereof must capture. At a conceptual level, Active Inference seems to capture the inherently contextual nature of cognitive engagement with the world. However, the Free Energy Principle cannot formalize this intuition due to its definition in terms of Dynamical Systems Theory, which comes with a well-defined space of possible states with no contextual properties. We describe the duality between agent-driven interaction and the construction of cognitive spaces which is implicit in the very concept of cognitive contextuality. In consequence, we emphasize the importance of the ontic regime called "Participatory Realism", and consider some ways that the FEP could integrate it.

Keywords: Active Inference · Free Energy Principle · Physics of cognition · Contextuality · Participatory realism · Open-ended evolution · Quantum information theory.

Introduction

The Active Inference Framework (AIF) is a recent computational theory of cognition [8, 26, 49], as well as a modeling framework for computational agents [52, 11, 10]. At its core, it gives a counterintuitive but powerful picture of how our minds work. According to the AIF, neuronal activity organizes so as to predict upcoming sensory signal, hence producing a coherent, dynamic perception of the world organized around the prior beliefs of the agent. Critically, proponents of the AIF argue that this predictive process also encompasses motor signal, underlying efficient and flexible motor control [25, 1, 26]. At its core, the AIF relies on one core mathematical result, the Free Energy Principle (FEP). The FEP

was initially formulated as a precursor of the AIF, *i.e.* a computational theory of the brain which holds that neural activity maximizes the Bayesian coherence at hierarchically nested scales [28, 27, 20]. It quickly evolved into the more ambitious attempt to formally ground this claim in the mathematical context of Dynamical Systems Theory (DST) [21, 22, 23], and integrate it to the broader context of mathematical physics [51, 24]. Successive works approached the question through different mathematical perspectives, and clarified the distinction between different subcases depending on the dynamical structures of the agent-environment system. But the core idea behind the FEP can be summarized in few words. In substance, any system which remains statistically separated from his environment develops and enacts an implicit cognitive model of its environment, embedded in the dynamics of the interaction.

However, the adequacy of the FEP to represent the processes underlying cognition has been called into question since its inception. Some lines of criticism leveraged against the FEP were defused by a careful discussion of the implications of the framework (*e.g* the Dark Room Problem [29], or the anti-representationalist argument [50]), or were made obsolete by later formal developments (*e.g.* the limitation of early formulations to strongly mixing systems [9]). To our knowledge, the main open line of criticism of the FEP (*e.g.* [12, 35, 36, 7]) tend to accentuate two core arguments. First, the FEP fails to reproduce adequately the asymmetry between agent and environment; second, the FEP fails to reproduce the open-ended evolution of biological and cognitive statespaces. The present discussion provides further context to this line of argumentation by showing the deep relation between those two arguments. Here, we attempt to address the question of the adequacy of the FEP from a meaningfully different (although complementary) angle. We will focus more specifically on the agency of cognitive beings in defining how they perceive and engage with the world, formalized in terms of contextuality. In particular, we hope that this axis will build on existing literature by showing how the two problems relate, and therefore facilitate further research intended to close the gap between the FEP and the ambitions behind its development.

Contextuality is a concept inherited from quantum mechanics, where it describes the property of states (*e.g.* position or momentum of a particle) which are not well-defined prior to their observation. Following [34], we propose to apply this concept to the construction of cognitive meaning by agents. Indeed, contextuality applies in general to properties that are observer-dependant. This applies to cognitive semiotics, regardless of whether the dynamics of cognition or the world an agent engages with are distinctly quantum-like. For example, where I see a book filled with carefully chosen and arranged symbol to convey a specific communicative intent, my cat may see a dangerous creature to be opposed by any means necessary. Hence, book-ness is contextual, or at the very least the cognitive meaning imprinted onto a given book is. We will see how this intuitive sense of contextuality maps onto its more formal meaning as non-commutativity between acts of observation (or, in the specific context we set for this discussion, cognitive engagement). Furthermore, we will argue that an

adequate model of cognitive engagement requires an explicit, non-reducible account of contextuality. Indeed, cognitive contextuality constitutes at its core an expression of agent-driven interaction, as it corresponds to a process where specific states become defined precisely because they are meaningful to the agent. But at the same time, it provides formal grounding to the open-ended evolution of cognitive meaning, as states which are brought about through cognitive contextuality become actual, causally efficient states of the world in the context of the interaction.

1 Contextuality in quantum physics

Contextuality, in the sense we call onto here, is a concept associated to the development of early quantum mechanics. The development of quantum mechanics was motivated in part by the strange fact that given the spectrum of electromagnetic emission by a black body, light energy seemed to travel by discrete packets of $E = h\nu$ (where h is Planck's constant, and ν is the frequency considered). By the 30s, physicists could account for that phenomenon through a fully coherent reformulation of physics in terms of *wavefunctions* and of the *operators* acting on them. This, however, entailed a major shift for the fundamental object treated by physics. Indeed, the evolution of quantum systems would not follow dynamics defined over an objective state of the system. Instead, it would evolve along a complex scalar field whose squared amplitude at a given point corresponds to the probability of finding the system at this very point if it was observed, and only collapse to specific states when observed. Einstein, Podolsky, and Rosen famously contested the coherence of the new theory by pointing out that collapse would violate locality (*i.e.* the fact that information can't travel faster than light), and quantum theory would need to be reduced to another theory where systems have objective states and dynamics that can't communicate faster than light [13]. The conjunction of those two conditions came to be known as *local realism*. However, the later development of Bell's theorem [3] showed that the empirical facts observed by quantum theory weren't compatible with local realism. Later, both Bell and a team formed by Kochen and Specker independently derived that realism itself was incompatible with the formalism of quantum mechanics [4, 46]. While the strategies for both demonstrations differ, we can explain this fact through simple observation about the basic objects of quantum mechanics.

In quantum theory, interaction is described through the application of operators acting over the Lebesgue space defined by the wavefunction. This construct generalizes the results of linear algebra for infinite-dimension spaces, like the space defined by each point where a wavefunction is defined, assuming that the wavefunction is normalizable (*i.e.* the integral of its square amplitude converges). Physical variables applying to a system are generally defined as linear application on the Lebesgue space of its wavefunction. To be precise, some operators over the wavefunction can be defined by their action on a complete orthonormal basis for the Lebesgue space; and have an eigenbasis where the action of the

operator on each component (called *eigenvectors*) reduces to multiplication by a scalar (called *eigenvalue*). Those operators can be associated to observables, *i.e.* physically meaningful, measurable variables like energy or position. The application of such operators to the system is understood to represent the measurement of its state along their specific eigenbasis. The output of the process is an eigenvalue of the corresponding operator, with probability proportional to the squared amplitude of the wavefunction projected onto the corresponding eigenvector (or integrated over the subspace corresponding to the eigenvalue). Additionally, the wavefunction is then projected onto the eigenvector or the subspace corresponding to the measured eigenvalue. Critically, two operators can only commute if they share the same eigenbasis. This means that in the general case, the final state of the system as well as the outcome of observation depend on the order in which a given series of measurement is applied [47]. In consequence, there isn't an intrinsic state of the system which is revealed by measurement; rather, measurement participates in the construction of the state of the system.

Commutativity & Quantum contextuality

Let us consider a simple qubit S characterized by the orthogonal eigenvectors $(|+\rangle ; |-\rangle)$. Its wavefunction is then defined by $a|+\rangle + b|-\rangle$, where $|a|^2 + |b|^2 = 1$.

One could want to measure whether the system is in state $|+\rangle$. They could use for that purpose the projector associated to the $|+\rangle$ state, $P_+ = |+\rangle\langle+|$. Using the bra-ket notation, $\langle+|$ is to be understood as the dual of $|+\rangle$. It is the linear application that sends $|\cdot\rangle$ to its scalar product with $|+\rangle$, *i.e.* $\langle+| : x \to \langle+|x\rangle$. Using the identities $\langle+|+\rangle = \langle-|-\rangle = 1$ and $\langle+|-\rangle = \langle-|+\rangle = 0$; it is straightforward that $P_+|+\rangle = |+\rangle$ and $P_+|-\rangle = 0$, so P_+ has for eigenstates $(|+\rangle ; |-\rangle)$ and for eigenvalues $(1; 0)$. Therefore, the corresponding measurement could output 1 and update the system in the state $|+\rangle$ with probability $|a|^2$; or it could output 0 and update the system in the state $|-\rangle$ with probability $|b|^2$.

On the contrary, one could want to measure whether S is in state $\frac{|+\rangle+|-\rangle}{\sqrt{2}}$. The associated projector is $P_1 = \frac{(|+\rangle+|-\rangle)(\langle+|+\langle-|)}{2}$. Again, it is straightforward that this projector has for eigenstates $(\frac{|+\rangle+|-\rangle}{\sqrt{2}}; \frac{|+\rangle-|-\rangle}{\sqrt{2}})$ and for corresponding eigenvalues $(1; 0)$. In consequence, the associated measurement would either output 1 and update the system in the state $\frac{|+\rangle+|-\rangle}{\sqrt{2}}$ with probability $\frac{|a+b|^2}{2}$; or output 0 and update the system in the state $\frac{|+\rangle-|-\rangle}{\sqrt{2}}$ with probability $\frac{|a-b|^2}{2}$

As P_+ and P_1 do not share an eigenbasis, they should not commute. In other words, their application in different order should give different results. Using the eigenmatrix representation of P_+ and P_1 in the eigenbasis $(|+\rangle ; |-\rangle)$, it is straighforward to verify that P_+P_1 and P_1P_+ respectively project the system on the eigenbasis $(|+\rangle ; \frac{|+\rangle+|-\rangle}{\sqrt{2}})$ and $(\frac{|+\rangle+|-\rangle}{\sqrt{2}}; |-\rangle)$.

Therefore, the state of the qubit S depends on the order in which the successive measurements are applied.

A fortiori, we cannot describe the outcome of the experiment as an "objectively real" state which preexists the process of observation, and is simply revealed through measurement. The state of the qubit is brought about by the act of observation, and more specifically by the context set by the associated operator. This property is what we call *quantum contextuality*.

Nearly all experts agree today that contextuality constitutes a fundamental property of quantum phenomena, as is entailed by mainstream quantum formalism. A thornier debate lies in whether contextuality derives from the structure of quantum statespaces, or quantum statespaces themselves derives from contextual properties. As of 2013, roughly identical proportion of active researchers in the foundation of quantum mechanics prefer either interpretation, with a slightly higher proportion prefer a mix of the two [53]. At one extreme of the distribution is the Copenhagen interpretation, which can be characterized as direct realism about the mathematical model described above. At the other is quantum Bayesianism (qBism), which holds on the contrary that quantum wavefunction are only a manifestation of the Bayes-optimal information that the broader world holds about the states of a given system given past interactions [38]. qBism is best supported by a series of mathematical results which show that most of quantum formalism can be reconstructed only from the constraint of Bayesian coherence [32, 33]. Arguably, it is empirically supported by the realization of Wheeler's delayed choice experiment [43], which demonstrates that the act of measuring either particle-like or wave-like property of a photon can determine whether the photon travelled as a particle or a wave *prior to the measurement*. This entails either retro-causality (a direct violation of the universally accepted principle of causality), or the more likely interpretation that the particle or wave nature of the photon is not well defined prior to its observation - as was initially proposed in [56]. To be clear, qBism emphatically does not deny the existence of physical reality. Rather, it holds that contextuality is a fundamental trait of physical reality, a thesis they characterize as Participatory Realism [31].

Participatory Realism entails a highly non-trivial view of the nature of the universe, and the role that time plays in it. The standard picture accepted by most physicists is that the universe is characterized by a fundamental underlying reality, *i.e.* a collections of objects, structures, and states (typically a collection of wavefunctions and field embedded in the fabric of spacetime) from which all physical reality stems. In this picture, time plays the role of a specific dimension characterized by the property of causality, meaning that past events but not future events can influence present events (given the constraints of general relativity, and outside of black holes). However, from a mathematical perspective, it conserves the structure of a simple dimension along which the trajectories of existing objects are defined according to specific rules of evolution. On the contrary, Participatory Realism holds that existing objects and properties are actively brought about by the very act of their observation. Accordingly, the collection of objects that exist at a given point in the universe is characterized

by the specific history of observation acts which brought the observer to that point. Properties such as the energy or momentum of a specific particle are not well-defined, prior properties of a given wavefunction; on the contrary, the structure of the wavefunction is brought about by the act of observing its energy or momentum. In consequence, time plays a constructive role in the definition of physical entities, in the sense that the collection of physical entities which are taken to exist is a function of the trajectory of the observer in time (the specific series of observations that they happen to make). Again, this does not entail a negation of the very concept of physical reality, or a strong form of anthropocentrism. Indeed, measurement can be simply operationalized as a non-commutative interaction between arbitrary objects, which reify the measured property in the context of their interaction and entails time-irreversibility. However, it does entail that the collection of objects within a universe is not a prior physical property, but is actively brought about by the history of interactions within that specific universe.

2 Contextuality in cognitive science

Several authors have drawn an explicit link between the ontic regime of Participatory Realism, as formalized in the context of qBism, and the nature of perception and cognition [30, 34, 37]. The argument can be summarized as follows. The constraints of online perception-action regulation do not afford a full top-down representation of all possible states in the world followed by an explicit inference of external states & planification of motor command. First, the information that an agent gathers through their senses is extremely insufficient for this purpose; second, the computational and memory cost of this calculation is much beyond what any biological system can do. What agents can do instead is to "ask questions" to the world through a selective opening to sensory stimuli which underlies the semiotics of their engagement with the world [34]. The behavioral regulation underlying those questions can be very rigid, down-to-earth, and simple, like chemotaxis; it can be very complex, abstract and contingent, like financial audit. But by definition, they always entail the rules of engagement that the agent applies in specific situations and toward specific kinds of objects; and implicitly or explicitly, it underwrites what kinds of things exist from the perspective of the agent (*i.e.* its cognitive semiotics). For example, a given chemotaxic organism may experience sugar as [good, proceed] while they experience toxins as [bad, avoid]. The common thread of those conceptions is that, by the very existence of regulation mechanisms, agents *project meaning* onto the world by defining what type of stimuli is relevant from the perspective of the agent and what for. As noted by [18], this property places contextuality at the very core of cognition. What an agent sees in the world - more precisely, what the world *is to the agent* in semiotic terms - becomes dependent on the specific context of observation. Here, the "context of observation" is to be understood as the implicit meaning embedded in the specific dynamics underlying perception-and-action, and hence the cognitive semiotics deployed by the agent.

The question we seek to address in this section is as follows: is cognition indeed contextual in the strict sense deployed in quantum theory, and under which condition ? To reaffirm the definition we are working with, we call *contextuality* or *observer-dependency* the property of a system whose "intrinsic", causally efficient internal states are a function of the way they are observed. For the reason exposed in box [Quantum contextuality], a qubit is a contextual object; because it can be accessed without modification, a bit is not. One may intuit that because cognition is not a quantum system [4], and the world it engages with is not a quantum system, their interaction cannot display contextuality. This would be overlooking the fact that, unlike in the prototypal case of classical information theory, the agent and the world do not typically have a prior agreement on the data format they communicate with. In other words, the terms of interaction are not typically defined *a priori*, as the agent actively produces and projects meaning. For example, what one learns by engaging with a book depends critically on whether they are literate. One would not end up in the same state whether they learn to read and then engages with a book; or engages with the same book and then only learns to read. Hence, the acts of [learning to read] and [engaging with a book] do not commute, and a system which may engage in either activity displays contextuality at the adequate scale of description. The same argument applies trivially wherever a given act produces irreversible changes in dynamics of the agent's cognitive activity, *i.e.* sets the context for further engagement in the world. Critically, it also appplies at evolutionary scale (*e.g.* emergence of affordances [44]) or behavioural scale (*e.g.* priming or framing effects, perceptive decision [6, 42]).

This argument can be formalized further using the formalism of [17] (see also [16, 15, 14]). Assuming an interaction between an agent A and an environment E, the symmetry of the interaction is broken when the agent A develops a memory of the history of interaction. This memory can be mediated by concept learning, enabling the agent to understand the world through types which all come with their specific rules for engagement. More generally, it stands for any and all forms of causal role that specific observations play in the internal dynamics of the agent. Thermodynamically, the maintenance of this memory entails the expenditure of free energy by the agent, and therefore the implicit distinction between "meaningful" states (those that are observed and integrated in the system's memory) and "meaningless" states (those that are "burnt" as free energy in order to power the process of writing into the memory) [5]. Due to the intrinsic irreversibility of thermal dissipation, we can derive a further consequence of this argument. If observation (when it is to be causally relevant) entails the dissipation of free energy, it also produces an irreversible constraint over the space of states of the world an agent can observe. Hence, there exist series of observations that do not commute, and this process entails contextuality. To be clear, what

[4] Although we should note that key authors in our argument consider the topic open to debate, see *e.g.* [19]

[5] Note that a similar argument is also developed in Tegmark (2012)'s attempt to ground the notion of entropy on quantum cosmology.

we describe here is not an auxiliary byproduct of cognitive engagement with the world. In the general case, cognitive engagement with the world entails memory in the sense defined above. Furthermore, the categories that cognitive agents may develop are mutually incoherent. One given organism, at a given time and in a given context, may only experience sugar as [good, proceed] or [bad, avoid]. In a layman's term, we may claim that contextuality is fundamentally tied to the fact that some systems come to develop judgements on which states of the world matter to them and which do not, in a way that is determined by their own activity rather than prior physical laws.

From this ground, we may revisit the role of time in cognitive engagement. If we are to take seriously the active construction of cognitive semiotics by biological agents, time is not simply a dimension along which the (predefined) dynamics of agent-environment interaction unfold. On the contrary, the cognitive semiotics of an agent develop through their specific history of cognitive engagement with the world, and feed back onto the definition of their dynamics. To take back the example of literacy, the dynamics of one's engagement with a book are not the same before or after learning to read. Hence, the active definition of their world by a given cognitive agent entails a strong form of irreversibility, not only in the thermodynamics of the process of cognitive engagement itself, but also in the very definition of the terms in which the agent understands and engages with the world. Rather than a linear dimension, time takes the form of a complex topology along which specific observables can be reified or discarded by a given agent. The relation between time and conscious experience is a well studied topic in continental philosophy, especially in the phenomenological tradition [5, 41]. However, these intuition lack to our knowledge a thorough formal grounding. While we do not propose such an account here, we provide the formal intution that contextuality is a key property to ground the construction of cognitive meaning, and hence the emergence of time from the intrinsic irreversibilty of cognitive agency.

3 Active Inference and contextuality

At face value, the framework of Active Inference appears to be an excellent fit to describe the contextuality of human cognition. Under Active Inference, the dynamics of cognitive agents flow so as to approximate optimal Bayesian inference (see box [Bayesian inference]) about the states of the world, given the system of prior beliefs they have about the world. Its core specificity, as compared to earlier models of cognition such as good old fashioned Reinforcement Learning, is the postulation of a fundamental unity between perception and action. Under Active Inference, an agent anticipates at the same time their sensory and motor states, and the prediction they generate for motor states plays the role of motor command [25, 1, 26]. The core constraints of Bayesian coherence then applies to both their active and sensory states, so that the agent attempts to update adaptively prior beliefs they can change while conserving prior beliefs they can't. Under this modeling framework, core constraints over the continued existence of

cognitive agents are understood as particularly robust or precise beliefs about their self-identity. We can say that under Active Inference, an agent enacts a model of themselves [50], a process usually described as "self-evidencing" [40]. This argument, which may appear like a simple modeling trick, fundamentally reframes the problem of cognitive engagement with the world. Indeed, classical cognitive architectures would need to explicitly represent the world and compute *a priori* motor commands to engage with the world - an extremely costly process. On the contrary, an Active Inference agent can fully leverage the power of the context set by its exteroceptive, interoceptive, and proprioceptive states to adequately anticipate their upcoming actions, and then implement adaptive sensorimotor regulation at an extremely reduced cost [39].

This natural fit between Active Inference and cognitive contextuality comes into question when we consider the mathematical grounding offered by its proponents, the *Free Energy Principe* (FEP) [21, 22, 23]. In its simplest form, the FEP can be explained as follows. Be a stochastic dynamical system that verifies a partition between three subspaces (A, B, E) such that the states of A and E are conditionally independent given those of B, *i.e.* $p(a|b, e) = p(a|b)$ and $p(e|b, a) = p(e|b)$ for all (a, b, e) in (A, B, E) (as well as some other technical conditions not addressed here). Intuitively, this conveys the idea that all information about the interaction between an agent A and its environment E is mediated by statistical effect on the so-called Markov Blanket B. Then, there exist an information geometry which relates every state of A to a distribution over states of E $q_a(e)$ which implementes a belief-like distribution over external states. This information geometry can be taken to operationalize the relation between the internal states of the agent and the (implicit) Bayesian beliefs it holds about the external world. Furthermore, the flow of the most likely states of A act so as to minimize the variational free energy of this distribution given states of B [48]. This means that, given the defining assumptions of the FEP, the agent acts so as to optimize the Bayesian coherence of belief-like distributions entailed by the information geometry. Extensive work has been undertaken to generalize the result to dynamical trajectories rather than single states, and to clarify the distinction between agents verifying different coupling conditions [22, 24]. In particular, it was demonstrated that when the boundary B includes "active states" which are not directly influenced by states of E and do not directly influence states of A, the belief-like distribution for which the agent's internal flow maximizes Bayesian coherence includes those active states. In other words, the FEP successfully explains the *a priori* counter-intuitive behaviour of cognitive agents under Active Inference based on very simple physical assumption, reducing in essence to the individuation of the agent from its environment.

Yet, this extremely powerful result suffers from its material grounding. Indeed, the FEP is by definition constructed in the context of Dynamical Systems Theory. This is precisely the reason for its strength, as DST formalizes with maximal generality the idea of an object with a well-defined state embedded in a well-defined space, and which evolves along well-defined rules. Formally speaking, a dynamical system is typically defined by a function $\Phi : (TX) \to X$

such that $\Phi(0;x) = x$ and $\Phi(t2;\Phi(t1;x)) = \Phi(t1+t2;x)$; where X is the "phase space" where the system evolves and T is a time-like space. Yet, because of the same rule that enforces the coherence of the flow, a dynamical system cannot be contextual: for any two evolution operators $\Phi_1 : \Phi(t_1;)$ and $\Phi_2 : \Phi(t_2;)$, it is straightforward that $\Phi_1\Phi_2 = \Phi_2\Phi_1 = \Phi(t_1+t_2,)$. This is assuming the time dimension is itself commutative, which instantiate the Cartesian intuition of time as a well-defined dimension equipped with a natural measure of distance (to wait 5 seconds then 3 is the same as to wait 3 seconds then 5). In the context of the FEP, this entails the necessary attraction of the system toward a predifined attracting distribution $p(A, B, E)$, which is at the core of the formalism. Indeed, the information geometry of the system is defined by reference to the attracting distribution, and it is therefore fully defined by the very definition of the system as a dynamical system. As seen above, it is precisely this information geometry which describes the emergence of implicit Bayesian belief about the world in the FEP, and therefore grounds cognitive meaning. Hence, the cognitive meaning (and the modes of engagement which it corresponds to in the context of our discussion) described by the FEP is well-defined and determined *a priori*, rather than a contextual product of the act of observation. Only the situation of the agent within the cognitive meaning defined by the information geometry may evolve in time, as is entailed by the definition of the dynamical flow. Hence, the contextuality of cognition is not accounted for by the FEP, neither in its intuitive nor formal aspects.

A formal avenue for the resolution of this problem seems to emerge when we consider a non-commutative time dimension. Indeed, the evolution of the system may then be framed to represent a history of irreversible events in the construction of the cognitive meaning it understands the world with, in the manner discussed in section 2. However, we must emphasize that this very assumption would dissolve much of the intuition provided by the FEP. Indeed, the core reason why the FEP is so important as a formal model of cognition is that it can reconstruct non-trivial traits of cognitive agency (Active Inference) from very basic physical assumptions about the cognitve system, namely the assumption that it exists as a dynamical system evolving in (cartesian) time. To assume the system evolves in a non-commutative time dimension, aimed to represent the contingent history of the very system it describes, would fundamentally invert this movement by building the framework on the basis of explicit, non-trivial assumptions about the physical nature of cognitive agency. In other words, a contextual reformulation of the FEP would explain the physical properties of the construction of cognitive semiotics from an ontology explicitly grounded on a conceptual account of cognitive agency, rather than explaining the cognitive properties of agent-environment interaction from a basic physical ontology. However, we argue this reframing of the FEP is a necessary step toward an explicit formalization of the constructive aspect of cognitive agency. As a counterpart of the increased ontological load of this framework, it would provide meaningful formal insight in the contextual evolution of physical statespaces, as is evoked (but not fully formalized) by qBism. Overall, the relevance of this approach can

only be assessed retrospectively, when we can compare the formal properties of a contextual reformulation of the FEP against those of existing physical formalism and the classical formulation of the FEP.

4 Conclusion

In this article, we have argued that the Free Energy Principle fails to reproduce a fundamental element of cognition, contextuality. Our argument proceeds in three times. First, we have explained the conceptual background and the formalism underlying the notion of contextuality in its original context, quantum mechanics. We emphasize that, while contextuality is a fundamental hallmark of quantum system, it does not require quantumness *per se*. Rather, it constitutes the expression of observer-dependancy in the definition of the target system, and therefore the lack of a predefined state prior to observation. Second, we have argued that contextuality indeed constitutes a fundamental aspect of cognition. In our development, we argued it is constitutively equivalent to the projection of meaning onto the world by a cognitive agent. Therefore, contextuality would be a fundamental condition of the construction of a perceptive field an agent can engage with, and hence of cognition itself. Third, we have discussed how the FEP fails to account for cognitive contextuality due to its dynamical systems-theoretic settings. By construction, DST supposes the coherent definition of the state in which a system is at any point in time, and is therefore incoherent with the contextuality of those states. This line of argumentation provides further context to prior discussion over the adequacy of the FEP as a formal theory of life and cognition. Indeed, cognitive contextuality constitutes at its core an expression of agent-driven interaction, as it corresponds to a process where specific states become defined precisely because they are meaningful to the agent. But at the same time, it provides formal grounding to the open-ended evolution of cognitive meaning, as states which are brought about through cognitive contextuality become actual, causally efficient states of the world in the context of the interaction.

To be clear, we do not deny that the Free Energy Principle or Active Inference constitute useful developments in mathematical physics and cognitive science respectively. Our view is rather that the rich picture of cognitive engagement drawn by Active Inference is not well grounded in the mathematics of the FEP, requiring further mathematical developments. By our present account of cognitive agency (and perhaps most importantly by the account proposed by Active Inference), the world perceived by the agent is actively produced by its own activity. This occurs notably through the enaction of causal constraints existing over its attention, which bring the agent to "bundle" collections of sensory and active states into perceptive concepts [37]. While in some instance (*e.g.* nematode cognition) we may perhaps discard the unfolding nature of concept creation as dynamically unsignificant, it is clearly not the case for human cognition. It should be beyond debate that if we agree to understand specific actions (*e.g.* words) as communicative acts embedded with abstracted meaning, this changes

concretely the perceptive space of possibilities we have access to, and this changes in kind the dynamics of our cognitive engagement with the world. The same goes for the construction of languages, symbols, and institutions throughout human history. However, as dynamical systems theoretic object are by definition embedded in a given set-theoretic space, they do not afford the emergence of novel possibilities through system evolution. In the context of the FEP, this translates into the definition of the external states the agent can perceive as a fixed and predefined collection of external states [35], without possibility for change or *a fortiori* for feedback on the agent's intrinsic dynamics. To the best of our knowledge, no mathematical framework currently describes the way from "bare" cognitive dynamics to the contextual definition of perceptive states and back. With the present discussion, we hope to have made a coherent case that this constitutes an important goal, and that the study of cognitive contextuality is a productive ground to do so.

Three lines of formal research seem of particular interest in that perspective. All have in common that they diverge significantly from the mathematical background of set theory, which is structurally reliant on the pre-definition of the space in which a system is defined [45]. First, some preliminary work points to the re-definition of the Free Energy Principle in the context of topos theory [2], a competing foundational theory of mathematics which embeds local topology in the very definition of individual states. This provides some grip on the notion of "where" a statement is verified, and therefore on cognitive contextuality. However, we should note that topos theory was specifically constructed to verify the structure of a *closed Cartesian category*, which means in a layman's terms that its mathematical structure allows the non-perturbative access to information (*contra* contextuality). Second, some authors have attempted to directly describe Active Inference in terms of category theory [54, 55], a mathematical discipline describing the structural relations within mathematical theories. In this context, Active Inference acquires the more general structure of a *closed monoidal category*, which accounts for perturbative access to information and therefore for a stronger sense of contextuality. Third, and perhaps most importantly, a small community of researchers have tried to formalize in category-theoretic terms the basic argument of qBism (as per section 1, the quantum theory which centers on the property of contextuality *per se* as the basic block of quantum theory), and have re-formalized the FEP within that context [16, 15, 14]. While mathematically abstract, these attempts at reframing the FEP matter pragmatically due to their potential capacity to account for the ontic regime we've called Participatory Realism, and therefore describe the agent-driven and open-ended nature of cognitive meaning.

Acknowledgments. This work was financed by the XSCAPE project (Synergy Grant ERC-2020-SyG 951631). I would like to thank Maxime Trimolet and Iona Brenac for proofreading, as well as the Kairos collective for overall support while I conducted this research. We also would like to thank anonymous reviewers for their useful comments.

Disclosure of Interests. The author declares no competing interests.

Bibliography

[1] Adams, R.A., Shipp, S., Friston, K.J.: Predictions not commands: Active inference in the motor system. Brain Structure and Function **218**(3), 611–643 (May 2013). https://doi.org/10.1007/s00429-012-0475-5

[2] Albarracin, M., Pitliya, R.J., St. Clere Smithe, T., Friedman, D.A., Friston, K., Ramstead, M.J.D.: Shared Protentions in Multi-Agent Active Inference. Entropy **26**(4), 303 (Apr 2024). https://doi.org/10.3390/e26040303

[3] Bell, J.S.: On the Einstein Podolsky Rosen paradox* **1**(3) (1964)

[4] Bell, J.S.: On the Problem of Hidden Variables in Quantum Mechanics. Reviews of Modern Physics **38**(3), 447–452 (Jul 1966). https://doi.org/10.1103/RevModPhys.38.447

[5] Bergson, H.: Essai sur les données immédiates de la conscience. Félix Alcan (1911)

[6] Cervantes, V.H., Dzhafarov, E.N.: Snow queen is evil and beautiful: Experimental evidence for probabilistic contextuality in human choices. Decision **5**(3), 193–204 (2018). https://doi.org/10.1037/dec0000095

[7] Chollat-Namy, M., Montévil, M.: What drives the brain ? Organizational changes, FEP and anti-entropy (Apr 2024)

[8] Clark, A.: Surfing Uncertainty: Prediction, Action, and the Embodied Mind. Oxford University Press (Oct 2015)

[9] Da Costa, L., Friston, K., Heins, C., Pavliotis, G.A.: Bayesian mechanics for stationary processes. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences **477**(2256), 20210518 (Dec 2021). https://doi.org/10.1098/rspa.2021.0518

[10] Da Costa, L., Lanillos, P., Sajid, N., Friston, K., Khan, S.: How Active Inference Could Help Revolutionise Robotics. Entropy **24**(3), 361 (Mar 2022). https://doi.org/10.3390/e24030361

[11] Da Costa, L., Parr, T., Sajid, N., Veselic, S., Neacsu, V., Friston, K.: Active inference on discrete state-spaces: A synthesis. Journal of mathematical psychology (2020). https://doi.org/10.1016/j.jmp.2020.102447

[12] Di Paolo, E.D., Thompson, E., Beer, R.: Laying down a forking path: Tensions between enaction and the free energy principle. Philosophy and the Mind Sciences **3** (Jan 2022). https://doi.org/10.33735/phimisci.2022.9187

[13] Einstein, A., Podolsky, B., Rosen, N.: Can Quantum-Mechanical Description of Physical Reality Be Considered Complete? Physical Review **47**(10), 777–780 (May 1935). https://doi.org/10.1103/PhysRev.47.777

[14] Fields, C., Fabrocini, F., Friston, K., Glazebrook, J.F., Hazan, H., Levin, M., Marciano, A.: Control flow in active inference systems (Feb 2023). https://doi.org/10.48550/arXiv.2303.01514

[15] Fields, C., Friston, K., Glazebrook, J.F., Levin, M.: A free energy principle for generic quantum systems (Dec 2021)

[16] Fields, C., Glazebrook, J.F.: Information flow in context-dependent hierarchical Bayesian inference. Journal of Experimen-

tal & Theoretical Artificial Intelligence pp. 1–32 (Oct 2020). https://doi.org/10.1080/0952813X.2020.1836034

[17] Fields, C., Glazebrook, J.F.: Representing Measurement as a Thermodynamic Symmetry Breaking. Symmetry **12**(5), 810 (May 2020). https://doi.org/10.3390/sym12050810

[18] Fields, C., Levin, M.: How do Living Systems Create Meaning? Philosophies **5**(4), 36 (Dec 2020). https://doi.org/10.3390/philosophies5040036

[19] Fields, C., Levin, M.: Metabolic limits on classical information processing by biological cells (Mar 2021). https://doi.org/10.48550/arXiv.2103.17061

[20] Friston, K.: The free-energy principle: A unified brain theory? Nature Reviews Neuroscience **11**(2), 127–138 (Feb 2010). https://doi.org/10.1038/nrn2787

[21] Friston, K.: A Free Energy Principle for Biological Systems. Entropy **14**(11), 2100–2121 (Nov 2012). https://doi.org/10.3390/e14112100

[22] Friston, K.: A free energy principle for a particular physics. arXiv:1906.10184 [q-bio] (Jun 2019)

[23] Friston, K., Da Costa, L., Sajid, N., Heins, C., Ueltzhöffer, K., Pavliotis, G.A., Parr, T.: The free energy principle made simpler but not too simple. arXiv:2201.06387 [cond-mat, physics:nlin, physics:physics, q-bio] (Jan 2022)

[24] Friston, K., Da Costa, L., Sakthivadivel, D.A.R., Heins, C., Pavliotis, G.A., Ramstead, M., Parr, T.: Path integrals, particular kinds, and strange things. Physics of Life Reviews (Aug 2023). https://doi.org/10.1016/j.plrev.2023.08.016

[25] Friston, K., Daunizeau, J., Kilner, J., Kiebel, S.J.: Action and behavior: A free-energy formulation. Biological Cybernetics **102**(3), 227–260 (Mar 2010). https://doi.org/10.1007/s00422-010-0364-z

[26] Friston, K., FitzGerald, T., Rigoli, F., Schwartenbeck, P., Pezzulo, G.: Active Inference: A Process Theory. Neural Computation **29**(1), 1–49 (Nov 2016). https://doi.org/10.1162/NECO_a_00912

[27] Friston, K., Kiebel, S.: Predictive coding under the free-energy principle. Philosophical Transactions of the Royal Society B: Biological Sciences **364**(1521), 1211–1221 (May 2009). https://doi.org/10.1098/rstb.2008.0300

[28] Friston, K., Kilner, J., Harrison, L.: A free energy principle for the brain. Journal of Physiology-Paris **100**(1), 70–87 (Jul 2006). https://doi.org/10.1016/j.jphysparis.2006.10.001

[29] Friston, K., Thornton, C., Clark, A.: Free-Energy Minimization and the Dark-Room Problem. Frontiers in Psychology **3** (May 2012). https://doi.org/10.3389/fpsyg.2012.00130

[30] Froese, T.: Scientific Observation Is Socio-Materially Augmented Perception: Toward a Participatory Realism. Philosophies **7**(2), 37 (Apr 2022). https://doi.org/10.3390/philosophies7020037

[31] Fuchs, C.A.: On Participatory Realism. In: Durham, I.T., Rickles, D. (eds.) Information and Interaction: Eddington, Wheeler, and the Limits of Knowledge, pp. 113–134. The Frontiers Collection, Springer International Publishing, Cham (2017). https://doi.org/10.1007/978-3-319-43760-6_7

[32] Fuchs, C.A., Schack, R.: A Quantum-Bayesian Route to Quantum-State Space. Foundations of Physics **41**(3), 345–356 (Mar 2011). https://doi.org/10.1007/s10701-009-9404-8

[33] Fuchs, C.A., Schack, R.: Quantum-Bayesian coherence. Reviews of Modern Physics **85**(4), 1693–1715 (Dec 2013). https://doi.org/10.1103/RevModPhys.85.1693

[34] Guénin–Carlut, A.: On participatory realism. Kairos Journal (Oct 2022)

[35] Guénin–Carlut, A.: Physics of creation - Symmetry breaking, (en)active inference, and unfolding statespaces (Oct 2022). https://doi.org/10.31219/osf.io/68947

[36] Guénin–Carlut, A.: Strange things, statespace representation, and participatory realism - Comment on "Path Integrals, Particular Kinds, and Strange Things." by Friston et al (Oct 2023). https://doi.org/10.31219/osf.io/urz26

[37] Guénin–Carlut, A.: From the existential stance to social constraints - How the human mind becomes embedded in our social, cultural and material context (Jul 2024)

[38] Healey, R.: Quantum-Bayesian and Pragmatist Views of Quantum Theory. In: Zalta, E.N., Nodelman, U. (eds.) The Stanford Encyclopedia of Philosophy. Metaphysics Research Lab, Stanford University, winter 2023 edn. (2023)

[39] Hipólito, I., Baltieri, M., Friston, K., Ramstead, M.J.: Embodied skillful performance: Where the action is. Synthese (Jan 2021). https://doi.org/10.1007/s11229-020-02986-5

[40] Hohwy, J.: The Self-Evidencing Brain. Noûs **50**(2), 259–285 (2016). https://doi.org/10.1111/nous.12062

[41] Husserl, E.: On the Phenomenology of the Consciousness of Internal Time (1893–1917). Springer Science & Business Media (Dec 2012)

[42] Isharya, MS., Cherukuri, A.K.: Decision-making in cognitive paradoxes with contextuality and quantum formalism. Applied Soft Computing **95**, 106521 (Oct 2020). https://doi.org/10.1016/j.asoc.2020.106521

[43] Jacques, V., Wu, E., Grosshans, F., Treussart, F., Grangier, P., Aspect, A., Roch, J.F.: Experimental realization of Wheeler's delayed-choice GedankenExperiment. Science **315**(5814), 966–968 (Feb 2007). https://doi.org/10.1126/science.1136303

[44] Kauffman, S.A.: A World Beyond Physics: The Emergence and Evolution of Life. Oxford University Press (Apr 2019)

[45] Kauffman, S.A., Roli, A.: A third transition in science? Interface Focus **13**(3), 20220063 (Apr 2023). https://doi.org/10.1098/rsfs.2022.0063

[46] Kochen, S., Specker, E.P.: The problem of hidden variables in quantum mechanics. J. Math. Mech. **17**, 59–87 (1967)

[47] Messiah, A.: Quantum Mechanics. Courier Corporation (Feb 2014)

[48] Parr, T., Da Costa, L., Friston, K.: Markov blankets, information geometry and stochastic thermodynamics. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **378**(2164), 20190159 (Feb 2020). https://doi.org/10.1098/rsta.2019.0159

[49] Parr, T., Pezzulo, G., Friston, K.: Active Inference: The Free Energy Principle in Mind, Brain, and Behavior. MIT Press, Cambridge, MA, USA (Mar 2022)
[50] Ramstead, M.J., Kirchhoff, M.D., Friston, K.: A tale of two densities: Active inference is enactive inference. Adaptive Behavior **28**(4), 225–239 (Aug 2020). https://doi.org/10.1177/1059712319862774
[51] Ramstead, M.J., Sakthivadivel, D.A.R., Heins, C., Koudahl, M., Millidge, B., Da Costa, L., Klein, B., Friston, K.: On Bayesian Mechanics: A Physics of and by Beliefs (May 2022). https://doi.org/10.48550/arXiv.2205.11543
[52] Sajid, N., Ball, P.J., Parr, T., Friston, K.: Active inference: Demystified and compared. arXiv:1909.10863 [cs, q-bio] (Oct 2020)
[53] Schlosshauer, M., Kofler, J., Zeilinger, A.: A Snapshot of Foundational Attitudes Toward Quantum Mechanics. Studies in History and Philosophy of Science Part B: Studies in History and Philosophy of Modern Physics **44**(3), 222–230 (Aug 2013). https://doi.org/10.1016/j.shpsb.2013.04.004
[54] Smithe, T.S.C.: Open dynamical systems as coalgebras for polynomial functors, with application to predictive processing (Jun 2022). https://doi.org/10.48550/arXiv.2206.03868
[55] Tull, S., Kleiner, J., Smithe, T.S.C.: Active Inference in String Diagrams: A Categorical Account of Predictive Processing and Free Energy (Aug 2023). https://doi.org/10.48550/arXiv.2308.00861
[56] Wheeler, J.A.: The "Past" and the "Delayed-Choice" Double-Slit Experiment. In: Marlow, A.R. (ed.) Mathematical Foundations of Quantum Theory, pp. 9–48. Academic Press (Jan 1978). https://doi.org/10.1016/B978-0-12-473250-6.50006-6