



Delft University of Technology

Unbiased Active Inference for Classical Control

Baioumy, M.; Pezzato, C.; Ferrari, R.; Hawes, N.

DOI

[10.1109/IROS47612.2022.9981095](https://doi.org/10.1109/IROS47612.2022.9981095)

Publication date

2022

Document Version

Final published version

Published in

Proceedings 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)

Citation (APA)

Baioumy, M., Pezzato, C., Ferrari, R., & Hawes, N. (2022). Unbiased Active Inference for Classical Control. In *Proceedings 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 12787-12794). (IEEE International Conference on Intelligent Robots and Systems). IEEE.
<https://doi.org/10.1109/IROS47612.2022.9981095>

Important note

To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

Green Open Access added to TU Delft Institutional Repository

'You share, we take care!' - Taverne project

<https://www.openaccess.nl/en/you-share-we-take-care>

Otherwise as indicated in the copyright section: the publisher is the copyright holder of this work and the author uses the Dutch legislation to make this work public.

Unbiased Active Inference for Classical Control

Mohamed Baioumy^{*1}, Corrado Pezzato^{*2}, Riccardo Ferrari³, Nick Hawes¹

Abstract—Active inference is a mathematical framework that originated in computational neuroscience. Recently, it has been demonstrated as a promising approach for constructing goal-driven behavior in robotics. Specifically, the active inference controller (AIC) has been successful on several continuous control and state-estimation tasks. Despite its relative success, some established design choices lead to a number of practical limitations for robot control. These include having a biased estimate of the state, and only an implicit model of control actions. In this paper, we highlight these limitations and propose an extended version of the unbiased active inference controller (u-AIC). The u-AIC maintains all the compelling benefits of the AIC and removes its limitations. Simulation results on a 2-DOF arm and experiments on a real 7-DOF manipulator show the improved performance of the u-AIC with respect to the standard AIC. The code can be found at https://github.com/cpezzato/unbiased_aic.

I. INTRODUCTION

Active inference is a mathematical framework prominent in computational neuroscience [1]. It aims to explain decision-making in biological agents using free-energy minimization. Recent work has led to many schemes based on this framework, for an overview see [2]. Nevertheless, active inference for robot control is still a relatively new field. It is thus of particular importance to understand the limits of such methods. In this paper, we present an analysis of the drawbacks associated with specific design choices within the active inference controller (AIC). Additionally, we propose an extended and improved unbiased AIC (u-AIC), which we initially presented in [3] solely for fault-tolerant control. The effort in understanding the limits of the AIC for robot control started with our previous work [3], where we specifically analyzed the effect of joint state-estimation and control for fault tolerance. The current paper provides an in-depth analysis of the performance and limits of the AIC more generally (not just fault-tolerant control). Additionally, we formalize the u-AIC. This includes 1) a derivation of the new control architecture and possible extensions, 2) a proof of convergence for both the state estimation and the controlled system, and 3) the application of the u-AIC to a



Fig. 1: Manipulation of delicate items in a human-shared store environment.

real 7-DOF robot manipulator, which was previously only performed in simulation.

In this paper, we identify the drawbacks of the AIC and connect them to two root causes. First, in the AIC the estimated state (or *belief*) is biased toward the current goal/target state through a goal prior. This leads to reduced quality in state-estimation [4] but also influences precision learning (learning the precision/covariance of the sensory model) making the model parameters converge to a biased value [4], [5]. A range of issues also arises in fault diagnosis and fault-tolerant control as a result of the goal prior, such as false-positive fault detection [6], [3]. Second, the control action is not explicitly modeled as a random variable in the generative model of the AIC. This causes a range of issues on the control side. In real-world control applications, the integral control law typical of the AIC can cause saturation problems for the actuators when the agent fails to reach a target. This can happen for instance because of a collision. Other limitations from a control perspective are that the AIC does not naturally allow for the incorporation of feed-forward control signals [3] which could improve the overall performance of the system. Finally, the motion can be jerky in practice.

Interestingly, all these limitations are present in the AIC but are not intrinsically part of active inference as a general framework. One can design different controllers that stay true to the principles of active inference while mitigating the limitations. To this end, we present the u-AIC, originally introduced in [3]. We demonstrate the properties of the u-AIC in Section IV, as well as the convergence of both the state estimation and control which is still missing for the AIC [7].

The *contributions* of this paper are twofold: 1) we con-

^{*} Authors with equal contribution

¹Mohamed Baioumy and Nick Hawes are with the Oxford Robotics Institute, Oxford University, [mohamed, nickh]@robots.ox.ac.uk

²Corrado Pezzato is with the Cognitive Robotics Department, TU Delft, c.pezzato@tudelft.nl

³Riccardo Ferrari is with the Department of Systems and Control, TU Delft, r.ferrari@tudelft.nl

This research was partially supported by Ahold Delhaize. All content represents the opinion of the author(s), which is not necessarily shared or endorsed by their respective employers and/or sponsors.

cretely demonstrate the limitations of the AIC using both theoretical results and empirical demonstrations. 2) We formalize the u-AIC and show how it overcomes such limitations, providing proof of convergence and extensions to the controller. We believe that understanding the properties and the boundaries of the AIC is important for researchers that want to apply it for robot control, such that better and safer systems can be built using this framework.

A. Related work

Active inference has been proposed in neuroscience as a general theory of the brain [8] and was concisely reported in [9] using a notation and language closer to engineering and control. We refer to the active inference controller in [9] as AIC, from which the first real-world applications for robot control [7], [10] were derived. Recently, active inference has been applied for robot control on a broader variety of settings and tasks. This includes work on robust state-estimation [11], [12], adaptive control [7], [4], fault-tolerant control [3], [13], reinforcement learning [14], planning under uncertainty [15], [16], human-robot interaction [17], [18] and more. The recent survey in [2] provides a detailed overview of active inference for robot control to date.

Particularly interesting in the context of this paper, is that several recent approaches in robotics have taken inspiration from the free-energy principle and applied it to continuous control and state estimation. Recent works focused on chance-constrained active inference [19], LQG control [20] and model-predictive control [21].

However, while the area of application of active inference in engineering settings is continuously growing, works on in-depth analysis of the AIC from a control theoretic perspective are limited. Work in [22], [4] has shown the relationship between the AIC and PID controllers while the relationship between the AIC, LQR control, and Kalman filters is discussed in [23], [20]. Finally, in [6], [3], some limitations of the AIC were discussed limited to fault-tolerant control, which motivated the introduction of the u-AIC [3]. This work focuses on a thorough analysis of the limits of the AIC from a control point of view and proposes an extended version of the u-AIC to address them.

B. Structure of the paper

The remainder of this paper is organized as follows. After providing the necessary mathematical background in Section II, Section III details the limitations of the AIC when applied to robot control. Section IV presents the u-AIC scheme to overcome these limitations. In Section V the properties of the u-AIC and its relation with the AIC are highlighted. In Section VI the theoretical claims are validated in simulation and on a real 7-DOF Franka Emika Panda manipulator. Finally, we draw conclusions in Section VII.

II. PRELIMINARIES

In this section, we concisely report the AIC formulation as used in previous work (e.g. [7], [4], [24], [10]), which specified the AIC for robot control according to the theory

in [9]. In this section, we only report the crucial equations to understand the limitations of the AIC in Section III, an interested reader is referred to [7] for all the details.

A. The generative model

Active Inference considers an agent in a dynamic environment that receives observations \mathbf{y} about states \mathbf{x} at every time-step t . The generative model of the agent can then be expressed as:

$$p(\mathbf{x}, \mathbf{y}) = \underbrace{p(\mathbf{y}|\mathbf{x})}_{\text{observation model}} \underbrace{p(\mathbf{x})}_{\text{prior}}. \quad (1)$$

A visual representation of this model is presented in fig. 2 (right). The probability distribution $p(\mathbf{y}|\mathbf{x})$ has a mean $\mathbf{g}(\mathbf{x})$, which is a mapping from state to observation. The prior $p(\mathbf{x})$ has a mean $\mathbf{f}(\mathbf{x})$, which is a function that encodes the goal state μ_g . The agent aims to infer the posterior $p(\mathbf{x}|\mathbf{y})$ given a model of the agent's world. This means finding the belief μ over the state \mathbf{x} given the observations. This can be achieved by minimizing the so-called variational free energy. If all distributions in eq. (1) are Gaussian, the free energy becomes a sum of least square terms [7].

B. Free-energy

The AIC performs joint state estimation and control by minimizing the free-energy \mathcal{F} through a gradient descent scheme. \mathcal{F} is defined as [7]:

$$\mathcal{F}(\mathbf{y}, \mu) = \frac{1}{2} \sum_{i=0}^{n_d-1} \left[\varepsilon_y^{(i)\top} \Sigma_{y^{(i)}}^{-1} \varepsilon_y^{(i)} + \varepsilon_\mu^{(i)\top} \Sigma_{\mu^{(i)}}^{-1} \varepsilon_\mu^{(i)} \right] + K. \quad (2)$$

The free energy is a weighted sum of prediction errors up to a constant K resulting from the derivations [9]. The terms $\Sigma_{y^{(i)}}^{-1}$ and $\Sigma_{\mu^{(i)}}^{-1}$ are precision matrices representing the confidence about sensory input and internal beliefs. These can be seen as tuning parameters. The term n_d represents the number of derivatives considered in the control problem. If we assume as in most cases [7], [10], [3] that $n_d = 2$, then state estimation and control are performed on position and velocity. These quantities are internally represented as the beliefs $\mu^{(0)} = \mu$ for positions, and $\mu^{(1)} = \mu'$ for velocities. The terms $\varepsilon_\mu^{(i)} = (\mu^{(i+1)} - \mathbf{f}^{(i)}(\mu))$ and $\varepsilon_y^{(i)} = (\mathbf{y}^{(i)} - \mathbf{g}^{(i)}(\mu))$ are respectively the *state* and *sensory* prediction errors.

The function $\mathbf{g}(\mu)$ represents the mapping between states and sensory observations. In case of robot control with position and velocity sensors, this is the identity mapping, so $\mathbf{g}(\mu) = \mu$. The function $\mathbf{f}(\mu) = \mu_g - \mu$ specifies the desired evolution of the dynamics of the system. In this case, [7], the AIC will make the system behave like a first-order linear system with desired goal position μ_g . By definition [9], [7], it holds:

$$\mathbf{g}^{(i)} = \frac{\partial \mathbf{g}}{\partial \mu} \mu^{(i)}, \quad \mathbf{f}^{(i)} = \frac{\partial \mathbf{f}}{\partial \mu} \mu^{(i)}, \quad \mathbf{g}^{(0)} = \mathbf{g}, \quad \mathbf{f}^{(0)} = \mathbf{f}. \quad (3)$$

Finally, the terms $\Sigma_{\mu^{(i)}}$ and $\Sigma_{y^{(i)}}$ are diagonal covariance matrices. In the scalar case, these are simply represented as the variances σ_μ and σ_y .

C. State estimation

State estimation in the AIC is achieved by gradient descent on the free-energy [9], [25], [7] with the update rule:

$$\dot{\tilde{\mu}} = \frac{d}{dt}\tilde{\mu} - \kappa_{\mu} \frac{\partial \mathcal{F}}{\partial \tilde{\mu}}, \quad (4)$$

where $\tilde{\mu} = [\mu, \mu']$, and t refers to time. The term κ_{μ} is a tunable learning rate.

D. Control

In the AIC, the control input u is also computed through gradient descent on \mathcal{F} . As can be seen by eq. (2), however, \mathcal{F} does not depend on u explicitly. On the other hand, the actions indirectly influence \mathcal{F} by making the state evolve over time and thus changing the sensory output. We can compute the actions using the chain rule as [9], [7]:

$$\dot{u} = -\kappa_a \frac{\partial \tilde{y}}{\partial u} \frac{\partial \mathcal{F}}{\partial \tilde{y}} \quad (5)$$

where $\tilde{y} = [y, y']$, and κ_a is the tuning parameter. The term $\frac{\partial \tilde{y}}{\partial u}$, requires a forward dynamic model and is generally hard to compute in closed-form for non-linear systems. In most previous work [7], [10], [3], [4] such need has been obviated by introducing a linear approximation. The expression can alternatively be written as a sum for every sensor y in \tilde{y} . This relationship shows how the control action is directly related to the sensory inputs and the *number* of sensors as seen in the following expression:

$$\dot{u} \approx -\kappa_a \frac{\partial \mathcal{F}}{\partial \tilde{y}} = -\kappa_a \sum_y \frac{\partial \mathcal{F}}{\partial y}. \quad (6)$$

As an example, for a system with a position sensor y_q , and velocity sensor $y_{\dot{q}}$ this control law would be:

$$\dot{u} = -\kappa_a [\Sigma_{y_q}^{-1}(\mathbf{y}_q - \mu) + \Sigma_{y_{\dot{q}}}^{-1}(\mathbf{y}_{\dot{q}} - \mu')]. \quad (7)$$

III. LIMITATIONS OF THE AIC

In this section, we discuss the limitations of the AIC for robot control. We note that these limitations are not inherent in the active inference framework but rather in how the AIC is constructed. In Section III-A we address *Limitation #1*: in AIC the belief over the *current* state is biased toward the target the agent aims to reach by means of goal prior. This means that the agent's belief is never accurate, except when the target is reached. In Section III-B we address *Limitation #2*: the control action is not explicit in the generative model. This means, that one cannot minimize the free energy with respect to the actions directly and instead use the chain rule as in eq. (5), making assumptions about the linearity of the system. This can cause a saturation problem and does not allow for the incorporation of feed-forward control signals to improve performance.

A. Limitation #1: biased state estimation

Before formally analyzing the controller, we provide a visual explanation using factor graphs. In fig. 2, the generative model of the AIC in eq. (1) is depicted. Each random variable is represented by a circle, and each probability distribution by a black square. By inspecting this graph, we see that the state x is connected to two distributions. The distribution $p(y|x)$ moves the belief closer to the observed value y . This distribution is represented in the free-energy \mathcal{F} by the term $\epsilon_y^T \Sigma_y^{-1} \epsilon_y$ where $\epsilon_y = (y - \mu)$. This is considering, as commonly done, $g(\cdot)$ as the identity mapping.

The second distribution is the prior $p(x)$. This distribution is Gaussian with the function $f(\mu) = \mu_g - \mu$ as its mean. The function encodes the goal state μ_g . This term moves the belief μ towards the goal state. The belief of the AIC is thus always biased towards the goal state. This is by design. The benefit of this is that now we can also compute a control action that moves the agent to the goal (using eq. (6)). However, the drawback is that state estimation is inaccurate.

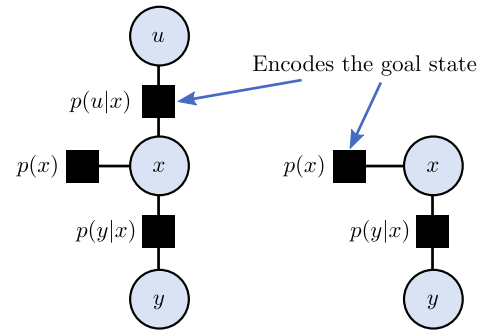


Fig. 2: Illustration of the u-AIC (left) and the AIC (right) for a one-dimensional problem. Circles indicate random variables. Black boxes indicate probability distributions.

1) Convergence to a biased belief: We will now analyze the convergence of the beliefs in the AIC. Let us consider a generic form of the free-energy, as in eq. (2). For the simplest linear and scalar case with $n_d = 1$ (i.e. one sensor and one actuator, with observable state), this expression can be reduced to:

$$\mathcal{F} = \frac{1}{2} \left[\frac{(y - g(\mu))^2}{\sigma_y} + \frac{(\mu' - f(\mu))^2}{\sigma_\mu} \right] + K \quad (8)$$

Let us consider the generative model of the state dynamics as a first order linear system with unitary time constant, so $f(\mu) = \mu_g - \mu$, and since the state is observable, so $g(\mu) = \mu$ [7]. At steady state, it holds that

$$\frac{\partial \mathcal{F}}{\partial \mu} = -\frac{(y - \mu)}{\sigma_y} + \frac{(\mu' - \mu_g + \mu)}{\sigma_\mu} = 0. \quad (9)$$

Additionally, when solving for μ we can show that

$$\mu = \frac{\sigma_\mu y + \sigma_y \mu_g - \sigma_y \mu'}{\sigma_\mu + \sigma_y}. \quad (10)$$

From eq. (10), one can notice that the belief μ is a weighted sum of the sensory measurement and the goal

state. The belief is thus always biased towards the goal. We demonstrate incorrect state estimation in simulation (section VI-A), which is particularly evident when collisions happen with the environment. The bias in the state estimation leads to two other issues. These are apparent when one tries to optimize model parameters through *precision learning* (or inverse covariance learning), and when applying the AIC for *fault tolerant control*. We expand upon these claims in the following sections.

2) **Biased precision learning:** Past work on robot control such as [7], [10] assumed that the precision σ_y^{-1} of the observation model is known. Intuitively, this quantifies the confidence about how noisy the sensor is, and can be determined before the deployment of the controller. However, for certain applications where sensory noise is uncertain, one might need to estimate the precision of the sensor online. This is the case in many robotics applications [26], [27].

There exists a body of work related to active inference which explores precision learning., e.g. methods such as Dynamic Expectation Maximization and generalized filtering [28], [29]. However, these methods do not perform control. When precision learning is done in conjunction with the active inference controller, we cannot estimate the true precision of the sensor. Work in [5], [4] shows how precision learning in the AIC can be used to find the optimal gains of the controller. In this context, however, the obtained precision of the sensor loses its physical meaning.

As explained earlier, the belief over the current state is biased, and this propagates to the precision of the system. Consider again the simplest linear and scalar case of the free-energy in eq. (8). The constant K can be expanded to include the variances as (see [9]):

$$K = \frac{1}{2} \ln \sigma_y \sigma_\mu.$$

Considering these terms time-varying, we can take the gradient with respect to the variances as:

$$\frac{\partial \mathcal{F}}{\partial \sigma_y} = -\frac{(y - \mu)^2}{2\sigma_y^2} + \frac{1}{2\sigma_y}. \quad (11)$$

Setting this expression to zero, we obtain:

$$\sigma_y = (y - \mu)^2. \quad (12)$$

The updated variance depends on μ which we showed is biased. Thus the precision will also be biased. This is because the agent is estimating the precision as the average square distance between the mean μ and every measurement y . If μ is biased, then the agent is estimating the variance around a value that is not the true mean. A special case is when the agent already starts at the goal. In that setting, μ would represent the true mean and the precision would be estimated around the true mean resulting in an accurate precision estimate. Finally, learning the precision of the observation model can be used in the context of control, *but* the obtained precision no longer represents the noise level of the sensor. Instead, it is a combination of the noise level, and how aggressive the controller will act, as shown in [21].

3) **Limitation in fault-tolerant control:** The AIC was used in [6] for fault tolerant control, where the main idea was that the sensory prediction errors in the free-energy could be used as residuals for fault detection, removing the need for more advanced residual generators. Despite the simplicity of the method for detecting and recovering from broken sensors, the use of prediction errors with biased state estimation can cause several false positives. This is explained in detail in [3], where a first version of the u-AIC has been proposed. We refer the reader to [3] for additional details.

B. Limitation #2: implicit modelling of actions

The control law for the AIC is computed using gradient descent on \mathcal{F} , as in eq. (5). Since the action is not explicitly modeled, one resorts to the chain rule. This introduces additional complexity. The term $dy/d\mu$ is generally hard to compute, and researchers approximated it by the identity matrix [5], [7]. Despite the promising results, linearizing the forward dynamics necessarily limits the performance in the presence of non-linear dynamics. Additionally, the control law is driven by sensory prediction errors weighted by the precision, see eq. (7), and it is de-facto an integrator. This causes three issues in practice.

First, the control law in the AIC is directly dependent on the observation. Eq. (6) shows that the control law is a sum of sensory prediction errors. This means that, if a system is not observable, it is not controllable. More specifically, under the assumptions explained in [4] and [5], the active inference controller is equivalent to a PI Controller i.e. PID with $D = 0$, a P gain of $\kappa_u \Sigma_y^{-1}$ and an I gain of $\kappa_u \Sigma_y^{-1}$. This assumes that the function $f(\mu) = (\mu_g - \mu)\tau^{-1}$ has a $\tau \rightarrow 0$. In that case, the belief will approach the goal state $\mu \rightarrow \mu_g$. In eq. (6), we see that the control law of the AIC is the sum of sensory prediction errors. Every term, in this case, is responsible for an error term in the PID control law. A position sensor is responsible for the I gain, a velocity sensor will result in a P gain, and an acceleration sensor results in a D gain.

Second, integral control has a few general drawbacks. For instance, if the goal cannot be reached due to a collision, the magnitude of the control action u will monotonically increase until saturation. This is shown in section VI for both simulated and real robots, where we point out that a vanilla implementation of the AIC leads to integrator windup.

Third, the control action cannot be naturally supplemented with a feed-forward signal. In case one has information about the system, designing a feed-forward signal to supplement the feedback controller consistently improves performance.

IV. UNBIASED ACTIVE INFERENCE CONTROLLER

A first version of the u-AIC has been introduced in [3] in the context of fault-tolerant control. In this section, we generalize, extend, and formally analyze the previously proposed u-AIC for generic control settings, and show how it overcomes the limitations of the AIC.

A. Derivation of the u-AIC

In this section, we describe the u-AIC as introduced in [3], to which an interested reader is referred for more details on the derivations of the following equations. Let us consider $\mathbf{x} = [\mathbf{q}, \dot{\mathbf{q}}]^\top$ and let us define a probabilistic model where actions are modelled explicitly:

$$p(\mathbf{x}, \mathbf{u}, \mathbf{y}) = \underbrace{p(\mathbf{u}|\mathbf{x})}_{\text{control}} \underbrace{p(\mathbf{y}|\mathbf{x})}_{\text{observation}} \underbrace{p(\mathbf{x})}_{\text{model prior}} \quad (13)$$

A visual representation of the u-AIC can be seen in fig. 2 (left). Note that with the u-AIC the information about the desired goal to be reached is encoded in the distribution $p(\mathbf{u}|\mathbf{x})$. This fundamentally removed the bias of the current state, as will be shown. In this paper, as in [4], we assume that an accurate dynamic model of the system is not available to keep the solution system agnostic and to highlight once again the adaptability of the controller.

The u-AIC aims at finding the posterior over states as well as the posterior over actions $p(\mathbf{x}, \mathbf{u}|\mathbf{y})$. Since the posteriors can be difficult to compute exactly, they are approximated using a variational distribution $Q(\mathbf{x}, \mathbf{u})$. We can make use of the mean-field assumption ($Q(\mathbf{x}, \mathbf{u}) = Q(\mathbf{x})Q(\mathbf{u})$) and the Laplace approximation, and assume the posterior over the state \mathbf{x} is Gaussian with mean $\boldsymbol{\mu}_x$ [30]. Similarly for the actions, the posterior \mathbf{u} is assumed Gaussian with mean $\boldsymbol{\mu}_u$. By defining the Kullback-Leibler divergence between the variational distribution and the true posterior, one can derive an expression for the free-energy \mathcal{F} as [3]:

$$\mathcal{F} = -\ln p(\boldsymbol{\mu}_u, \boldsymbol{\mu}_x, \mathbf{y}) + C \quad (14)$$

Considering eq. (13) and assuming Gaussian distributions, \mathcal{F} becomes:

$$\mathcal{F} = \frac{1}{2}(\boldsymbol{\varepsilon}_y^\top \Sigma_y^{-1} \boldsymbol{\varepsilon}_y + \boldsymbol{\varepsilon}_x^\top \Sigma_x^{-1} \boldsymbol{\varepsilon}_x + \boldsymbol{\varepsilon}_u^\top \Sigma_u^{-1} \boldsymbol{\varepsilon}_u + \ln |\Sigma_u \Sigma_y \Sigma_x|) + C, \quad (15)$$

The terms $\boldsymbol{\varepsilon}_{y_q} = \mathbf{y}_q - \boldsymbol{\mu}$, $\boldsymbol{\varepsilon}_{y_{\dot{q}}} = \mathbf{y}_{\dot{q}} - \boldsymbol{\mu}'$ are the sensory prediction errors respectively for position and velocity sensory inputs. The controller represents the states internally as $\boldsymbol{\mu}_x = [\boldsymbol{\mu}, \boldsymbol{\mu}']^\top$. The relation between internal state and observation is expressed through the generative model of the sensory input $\mathbf{g} = [\mathbf{g}_q, \mathbf{g}_{\dot{q}}]$. Position and velocity encoders directly measure the state, thus \mathbf{g}_q and $\mathbf{g}_{\dot{q}}$ are linear (identity) mappings.

Additionally, $\boldsymbol{\varepsilon}_u$ is the prediction error on the control action while $\boldsymbol{\varepsilon}_x$ is the prediction error on the state. The latter is computed considering a prediction of the state $\hat{\mathbf{x}}$ at the current time-step such that $\boldsymbol{\varepsilon}_x = (\boldsymbol{\mu}_x - \hat{\mathbf{x}})$. The prediction is a deterministic value $\hat{\mathbf{x}} = [\hat{\mathbf{q}}, \hat{\dot{\mathbf{q}}}]^\top$ which can be computed in the same fashion as the prediction step of, for instance, a Kalman filter. The prediction is approximated propagating forward in time the current state belief using the following simplified discrete time model:

$$\hat{\mathbf{x}}_{k+1} = \begin{bmatrix} I & I\Delta t \\ 0 & I \end{bmatrix} \boldsymbol{\mu}_{x,k} \quad (16)$$

where I represents a unitary matrix of suitable size. This form assumes that the position of each joint is thus computed as the discrete-time integral of the velocity, using a first-order Euler scheme. This approximation can be avoided if a better dynamic model of the system is available, and in that case, predictions can be made using the model itself. Finally, by choosing the distribution $p(\mathbf{u}|\mathbf{x})$ to be Gaussian with mean $\mathbf{f}^*(\boldsymbol{\mu}_x, \boldsymbol{\mu}_g)$, we can steer the system towards the target $\boldsymbol{\mu}_g$ without biasing the state estimation. This results in $\boldsymbol{\varepsilon}_u = (\boldsymbol{\mu}_u - \mathbf{f}^*(\boldsymbol{\mu}_x, \boldsymbol{\mu}_g))$.

In the u-AIC state estimation and control are achieved using gradient descent along the free energy. This leads to:

$$\dot{\boldsymbol{\mu}}_u = -\kappa_u \frac{\partial \mathcal{F}}{\partial \boldsymbol{\mu}_u}, \quad \dot{\boldsymbol{\mu}}_x = -\kappa_\mu \frac{\partial \mathcal{F}}{\partial \boldsymbol{\mu}_x}, \quad (17)$$

where κ_u and κ_μ are the gradient descent step sizes. The gradient on control can be computed as:

$$\frac{\partial \mathcal{F}}{\partial \boldsymbol{\mu}_u} = \Sigma_u^{-1}(\boldsymbol{\mu}_u - \mathbf{f}^*(\boldsymbol{\mu}_x, \boldsymbol{\mu}_g)) \quad (18)$$

B. Proof of convergence

For the simple 1-dimensional case, the expression of the free energy for the u-AIC can be reduced to:

$$\mathcal{F} = \frac{1}{2} \left[\frac{(y - \mu_x)^2}{\sigma_y} + \frac{(\mu_u - f^*(\cdot))^2}{\sigma_u} + \frac{(\mu_x - \hat{x})^2}{\sigma_x} \right] + K \quad (19)$$

The belief over the state $\boldsymbol{\mu}_x$ and over the control action $\boldsymbol{\mu}_u$ will converge when:

$$\frac{\partial \mathcal{F}}{\partial \boldsymbol{\mu}_u} = 0, \quad \frac{\partial \mathcal{F}}{\partial \boldsymbol{\mu}_x} = 0, \quad (20)$$

At steady state, it holds that $\mu_u = f^*(\mu_x, \mu_g)$. Considering this result, one can show that μ_x converges to

$$\mu_x = \frac{\sigma_x y + \sigma_y \hat{x}}{\sigma_x + \sigma_y}. \quad (21)$$

We can thus see that the control action converges to the function f^* , which can be chosen for instance as a PID controller. We can also show that belief over the state is a weighted average of the sensory measurement y and the prediction \hat{x} . Unlike the AIC, this prediction does not depend on the goal, see eq. (10).

C. Extensions of the u-AIC for control

The u-AIC can be extended for richer control by modifying the probabilistic model in eq. (13). This is because the control action \mathbf{u} is explicitly modeled as a random variable, and thus we can add prior probability distributions to it. This can be done in multiple ways to achieve different objectives.

1) **Adding a feed-forward controller (open-loop)**: So far the only term depending on the control action now is $p(\mathbf{u}|\mathbf{x})$; however, a prior $p(\mathbf{u})$ can be also added. In that case, the generative model would be:

$$\frac{1}{\alpha} p(\mathbf{u}) p(\mathbf{u}|\mathbf{x}) p(\mathbf{y}|\mathbf{x}) p(\mathbf{x}) \quad (22)$$

where α is a normalization constant. Note that, this denominator does not need to be computed explicitly. To

achieve state estimation and control, we simply need to minimize the free energy (which maximizes the likelihood of the model). We do not need to exactly compute the free energy or the likelihood.

This prior can encode a feed-forward signal (open-loop control law). Assuming the prior to be Gaussian with mean $f_{ol}(x)$ and variance σ_{ol} ('ol' stands for open-loop), we can show that the control law will converge to:

$$\mu_u = \frac{f_{ol}(\mu_x)\sigma_u + f^*(\mu_g, \mu_x)\sigma_{ol}}{\sigma_{ol} + \sigma_u}. \quad (23)$$

This is the weighted sum of the PID control law (after applying a filter) and the open-loop control law. Note previously, the value of σ_u did not matter, now the ratio between σ_u and σ_{ol} does contribute. Additionally, the expression for \mathcal{F} , would contain a quadratic term for the open-loop control law.

2) **Adding control costs:** Alternative to adding feed-forward control law, we might add a control cost. This can be achieved by adding the control prior $p_{cc}(u)$ with a mean of 0 and variance of σ_{cc} . This creates a quadratic term in \mathcal{F} of $(\mu_u - 0)^2/\sigma_{cc}$. This is equivalent to a quadratic control cost as seen in classical LQR controllers.

3) **Smoothing the control action:** The AIC often exhibits jerky motion. This can be mitigated in the u-AIC by adding a smoothing prior. Let u be a random variable referring to the current control action being executed. We then define u_p as the control action executed at the time previous time-step. A distribution $p(u|u_p)$ can be added to the generative model. This will add a control cost of $(\mu_u - u_p)^2/\sigma_p$ to \mathcal{F} . Minimizing this quantity nudges the control action u toward the value of the last control action u_p creating a smoothing effect. This quadratic loss is similar to approaches in Model-predictive control (MPC) [31].

A comparison of the standard AIC and u-AIC against MPC and impedance control can be found in [24]. Future work could address the comparison of the proposed extensions of the u-AIC against other classical controllers.

V. RELATIONSHIP BETWEEN THE AIC AND U-AIC

A. Convergence of beliefs

The AIC considers the relationship between states and observations without explicitly modeling the control actions. In classical filters, such as the Kalman filter, the prior comes from a prediction step that relies on the previous state and action. In the case of the AIC, the prior essentially predicts the agent to move towards the target. This results in a biased state estimate as seen in eq. (10). In contrast, the u-AIC has an unbiased belief over the state as seen in eq. (21). Note that both expressions are almost identical. The AIC converges to the weighted average between the sensory measurement and the goal. The u-AIC on the other hand converges to a weighted average between the sensory measurement and the predicted state (using a model or Euler integration). Note that, in the AIC, the **goal** state is encoded in the prior over the state $p(x)$, while in the u-AIC it is encoded in a separate distribution $p(u|x)$ (see fig. 2).

B. Control law

In the u-AIC, actions are explicitly modelled and the target state is encoded in the term $p(u|x)$. Explicit actions allow us to directly perform gradient descent on \mathcal{F} with respect to u . This is not possible in the general AIC case and the chain rule had to be utilized (see eq. (5)). The eventual control law of the AIC is also directly dependent on the measurement and the number of measurements eq. (6). Thus if part of the system is unobserved, it can not be controlled. The u-AIC on the contrary has a control law irrespective of the number of sensors eq. (21) and allows us to encode control costs, smoothing effects and feed-forward control signals (see eq. (23)).

C. General architecture

The general architecture (for state estimation and control) for the AIC and u-AIC is also different. In the case of the AIC, state estimation is dependent on the goal state. The goal is encoded in the prior, which biases the state estimation. The controller, on the other hand, is dependent on the (biased) belief and measurements. This is unusual in classical control, see fig. 3. A discussion on the role of modularity in AIC and the general control architecture can be found in [32]. The u-AIC on the other hand has a more traditional flow

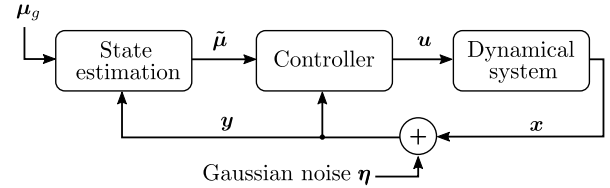


Fig. 3: Control diagram of the AIC

of information. State estimation requires the measurements y . Then the (unbiased) belief μ_x and the goal μ_g are fed into the controller which produces the control action. This is illustrated in fig. 4.

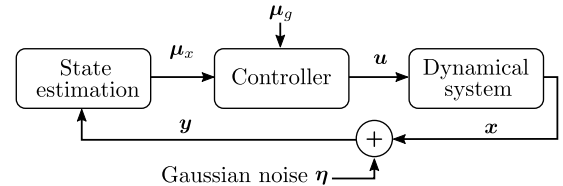


Fig. 4: Control diagram of the u-AIC.

VI. EXPERIMENTAL EVALUATION

We now showcase the limitations of the AIC explained in Section III and compare the performance with the u-AIC, both in simulation and in the real world.

A. Simulation

The simulation scenario is depicted in fig. 5. The robot has to reach a target in configuration space. During motion, we suppose that at a certain time a collision occurs which prevents the robot from moving further. We assume the

collision persists for $t_c = 3s$, then the robot is free to proceed. This allows us to show the incorrect state estimation and the overshoot due to the integral control law of the AIC. In the u-AIC, we observe none of these while maintaining similar performance.

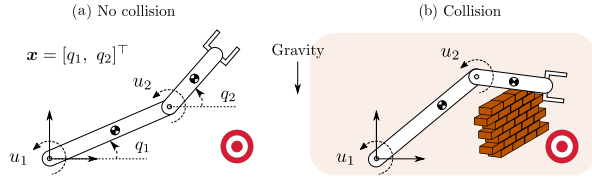


Fig. 5: Scenarios considered to illustrate the incorrect state estimation due to the bias towards the target (red dot). In (b) an obstacle occludes the way and blocks the arm.

The 2-DOF robot arm is equipped with position and velocity sensors y_q , $y_{\dot{q}} \in \mathbb{R}^2$ for the two joints affected by zero mean Gaussian noise, as in fig. 5. We define $y = [y_q, y_{\dot{q}}]^\top$. The states x to be controlled are set as the joint positions $q = [q_1, q_2]^\top$ of the robot arm. The simulation results for AIC and u-AIC are reported in fig. 6.

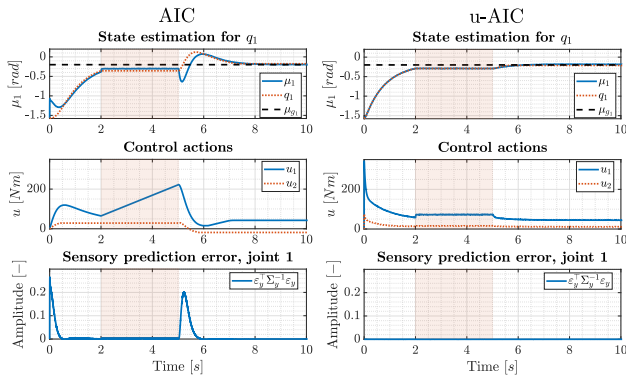


Fig. 6: Simulation results of a 2-DOF robot arm collision (orange area) for $t_c = 3s$. For readability, we only report the first joint since the behavior is similar for the second one.

The collision scenario in fig. 5 highlights a few limitations:

1) **Incorrect state estimation:** Let us consider the first row of the plots in fig. 6. When the AIC is not able to reach the desired target due to a collision blocking the path, the belief μ does not follow the trend of the real state x . The belief converges to a value between the sensory reading and the desired goal. For instance, at time $t = 4s$ the sensory reading is -0.355 [rad]. Given the current $\mu_g = -0.2$ [rad] as goal for the first joint, $\sigma_y = \sigma_\mu = 1$, and $\mu' = 0.053$ [rad/s] the belief converges to -0.304 [rad], which is in accordance with eq. (10). On the other hand, the u-AIC converges to the true state. For statistical significance, we ran 100 reaching tasks for each controller, with random blocking collisions of duration between $\in [1, 3]s$ at time $\in [0, 3]s$. Table I reports steady-state error (e_{ss}), settling time (t_s) after removing the collision, overshoot (os), and $RMSE$ between beliefs and joint positions in case of blocking the arm temporarily, averaged over the 100 trials. As can be seen,

Scenarios	e_{ss} [rad]	t_s [s]	os [%]	$RMSE$ [rad]
AIC	2.82e-05	3.70	0.12	0.042
u-AIC	0.0058	5.15	8.44	4.43e-4
AIC + coll.	6.07e-05	5.04	43.90	0.1695
u-AIC + coll	0.0053	5.86	11.65	4.92e-04

TABLE I: Simulation results of AIC and u-AIC without and with a random collision of random duration, averaged over 100 randomized reaching tasks.

while the AIC presents the lowest e_{ss} due to the prominent integration scheme, the $RMSE$ for state estimation is two orders of magnitude higher than for the u-AIC. Incorrect state estimation can also cause several false positives as described in detail in [3]. The sensory prediction errors for the AIC can in fact increase also due to collisions with the environment (see the last row of plots in fig. 6). The u-AIC is instead insensitive.

2) **Monotonic increase of control input:** The second row of the plots in fig. 6 displays the control action of one joint. During a collision, the control input computed by the AIC keeps increasing due to the integral nature of the control law employed (see eq. (7)). After the collision is removed, the controller necessarily overshoots. In the u-AIC, one can saturate only the integral term, and not the entire control law. On average, the AIC has four times the overshoot after a collision with a comparable settling time to the u-AIC.

B. 7-DOF Panda arm

On the real Panda arm, we compared 1) the reference tracking in configuration space, and 2) the collision behavior in terms of overshoot and commanded control actions resulting from holding the robot arm away from its desired set point. The behaviors of the controllers are best appreciated in the accompanying video¹.

1) **Reference tracking:** The performance in terms of reference tracking of a sinusoidal wave are reported in fig. 7. The AIC never converges to the reference trajectory, and this is independent of the initial conditions. We highlight this by plotting the response of the two controllers after running them for 4 seconds.

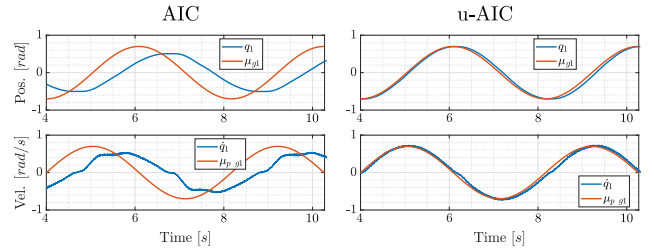


Fig. 7: Experiments with real Panda arm on reference tracking. For readability, we only report the first joint.

The AIC performs poorly in terms of tracking errors. This is due to three main reasons: 1) the AIC as defined in [9], [7] only allows for position reference through its generative

¹<https://www.youtube.com/watch?v=jI-zX8XvfgI>

model, while the u-AIC can perform position and velocity tracking; 2) a more aggressive tuning of the AIC would result in high overshoots in case of collision, compromising safety.

2) **Collision behavior:** The collision behavior is reported in fig. 8. The robot is commanded to keep an initial position while a person is pushing, pulling, and holding the robot. The AIC shows high overshoot which might be dangerous or damage delicate products such as fruits and vegetables in our setting. The u-AIC is instead well-behaved during interaction (see the attached video for a visualization of the results). The same behavior is observed when the robot is disturbed while performing a complicated trajectory.

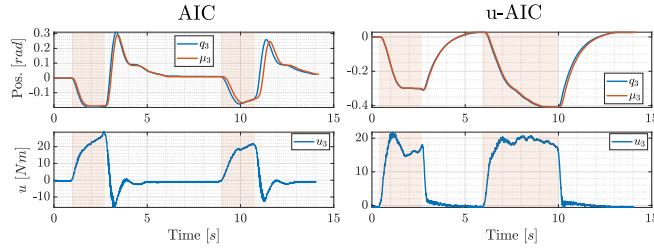


Fig. 8: Experiments with real Panda arm during unwanted interaction (orange area). For readability we only report the third joint, being the most affected one. Faster convergence to zero steady-state error for the u-AIC can be achieved through tuning of the integral action.

VII. CONCLUSION

In this paper, we discussed the fundamental limitations of the AIC for robot control and how the u-AIC overcomes them. These limitations arise from the fact that the state estimation is biased towards the goal, and that the control action is not explicitly modeled in the generative model. These cause degraded state estimation and can lead to large overshoots during human-robot interaction. We thoroughly discussed and extended the u-AIC providing a missing proof of convergence. We theoretically demonstrated the limitations of the AIC and provided experimental evidence of how the u-AIC overcomes them, both in simulation and in the real world with a 7-DOF robot manipulator.

REFERENCES

- [1] K. Friston, T. FitzGerald, F. Rigoli, P. Schwartenbeck, and G. Pezzulo, "Active inference: a process theory," *Neural computation*, vol. 29, no. 1, pp. 1–49, 2017.
- [2] P. Lanillos, C. Meo, C. Pezzato, A. A. Meera, M. Baioumy, W. Ohata, A. Tschantz, B. Millidge, M. Wisse, C. L. Buckley, *et al.*, "Active inference in robotics and artificial agents: Survey and challenges," *arXiv preprint arXiv:2112.01871*, 2021.
- [3] M. Baioumy, C. Pezzato, R. Ferrari, C. H. Corbato, and N. Hawes, "Fault-tolerant control of robot manipulators with sensory faults using unbiased active inference," in *European Control Conf. (ECC)*, 2021.
- [4] M. Baioumy, P. Duckworth, B. Lacerda, and N. Hawes, "Active inference for integrated state-estimation, control, and learning," in *Proc of IEEE Int. Conf. on robotics and automation (ICRA)*, 2021.
- [5] M. Baltieri and C. L. Buckley, "PID control as a process of active inference with linear generative models," *Entropy*, vol. 21, no. 3, 2019.
- [6] C. Pezzato, M. Baioumy, C. H. Corbato, N. Hawes, M. Wisse, and R. Ferrari, "Active inference for fault tolerant control of robot manipulators with sensory faults," in *1st Int. Workshop on Active Inference, ECML PKDD*, ser. Communications in Computer and Information Science, Springer, Ed., vol. 1326, 2020.

- [7] C. Pezzato, R. Ferrari, and C. H. Corbato, "A novel adaptive controller for robot manipulators based on active inference," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2973–2980, 2020.
- [8] K. J. Friston, "The free-energy principle: a unified brain theory?" *Nature Reviews Neuroscience*, vol. 11(2), pp. 27–138, 2010.
- [9] C. L. Buckley, C. S. Kim, S. McGregor, and A. K. Seth, "The free energy principle for action and perception: A mathematical review," *Journal of Mathematical Psychology*, vol. 81, pp. 55–79, 2017.
- [10] G. Oliver, P. Lanillos, and G. Cheng, "An empirical study of active inference on a humanoid robot," *IEEE Transactions on Cognitive and Developmental Systems*, 2021.
- [11] P. Lanillos and G. Cheng, "Adaptive robot body learning and estimation through predictive coding," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.
- [12] A. Meera and M. Wisse, "Free energy principle based state and input observer design for linear systems with colored noise," in *2020 American Control Conf. (ACC)*, 2020, pp. 5052–5058.
- [13] M. Baioumy, C. Pezzato, C. H. Corbato, N. Hawes, and R. Ferrari, "Towards stochastic fault-tolerant control using precision learning and active inference," in *IWAIF*. Springer, 2021.
- [14] A. Tschantz, B. Millidge, A. K. Seth, and C. L. Buckley, "Reinforcement learning through active inference," *arXiv preprint arXiv:2002.12636*, 2020.
- [15] L. Da Costa, N. Sajid, T. Parr, K. Friston, and R. Smith, "The relationship between dynamic programming and active inference: The discrete, finite-horizon case," *arXiv preprint arXiv:2009.08111*, 2020.
- [16] M. Baioumy, P. Duckworth, B. Lacerda, and N. Hawes, "On solving a stochastic shortest-path markov decision process as probabilistic inference," in *IWAIF*. Springer, 2021.
- [17] H. F. Chame, A. Ahmadi, and J. Tani, "A hybrid human-neurobotics approach to primary intersubjectivity via active inference," *Frontiers in Psychology*, vol. 11, p. 3207, 2020.
- [18] W. Ohata and J. Tani, "Investigation of the sense of agency in social cognition, based on frameworks of predictive coding and active inference: A simulation study on multimodal imitative interaction," *Frontiers in Neurobotics*, vol. 14, Sep 2020.
- [19] T. van de Laar, İ. Şenöz, A. Özçelikkale, and H. Wymeersch, "Chance-constrained active inference," *Neural Computation*, 2021.
- [20] T. van de Laar, A. Özçelikkale, and H. Wymeersch, "Application of the free energy principle to estimation and control," *IEEE Transactions on Signal Processing*, vol. 69, pp. 4234–4244, 2021.
- [21] M. Baioumy, M. Mattamala, and N. Hawes, "Variational inference for predictive and reactive controllers," in *BAIN-PIL workshop, ICRA*, Paris, France, 2020.
- [22] M. Baltieri and C. L. Buckley, "A probabilistic interpretation of PID controllers using active inference," in *Int. Conf. on Simulation of Adaptive Behavior*. Springer, 2018, pp. 15–26.
- [23] —, "On kalman-bucy filters, linear quadratic control and active inference," *arXiv preprint arXiv:2005.06269*, 2020.
- [24] C. Meo and P. Lanillos, "Multimodal vae active inference controller," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021.
- [25] K. Friston, K. Stephan, B. Li, and J. Daunizeau, "Generalised filtering," *Mathematical Problems in Engineering*, 2010.
- [26] W. Vega-Brown and N. Roy, "Cello-em: Adaptive sensor models without ground truth," in *2013 IEEE/RSJ International Conf. on Intelligent Robots and Systems*. IEEE, 2013, pp. 1907–1914.
- [27] T. Pfeifer, S. Lange, and P. Protzel, "Dynamic covariance estimation—a parameter free approach to robust sensor fusion," in *2017 IEEE International Conf. on Multisensor Fusion and Integration for Intelligent Systems (MFI)*. IEEE, 2017, pp. 359–365.
- [28] K. J. Friston, N. Trujillo-Barreto, and J. Daunizeau, "Dem: a variational treatment of dynamic systems," *Neuroimage*, 2008.
- [29] K. Friston, K. Stephan, B. Li, and J. Daunizeau, "Generalised filtering," *Mathematical Problems in Engineering*, vol. 2010, 2010.
- [30] K. Friston, J. Mattout, N. Trujillo-Barreto, J. Ashburner, and W. Penny, "Variational free energy and the Laplace approximation," *Neuroimage*, vol. 34(1), pp. 220–234, 2007.
- [31] L. E. Olivier and I. K. Craig, "Fault-tolerant nonlinear mpc using particle filtering," *IFAC-PapersOnLine*, 2016.
- [32] M. Baltieri and C. L. Buckley, "The modularity of action and perception revisited using control theory and active inference," *arXiv preprint arXiv:1806.02649*, 2018.