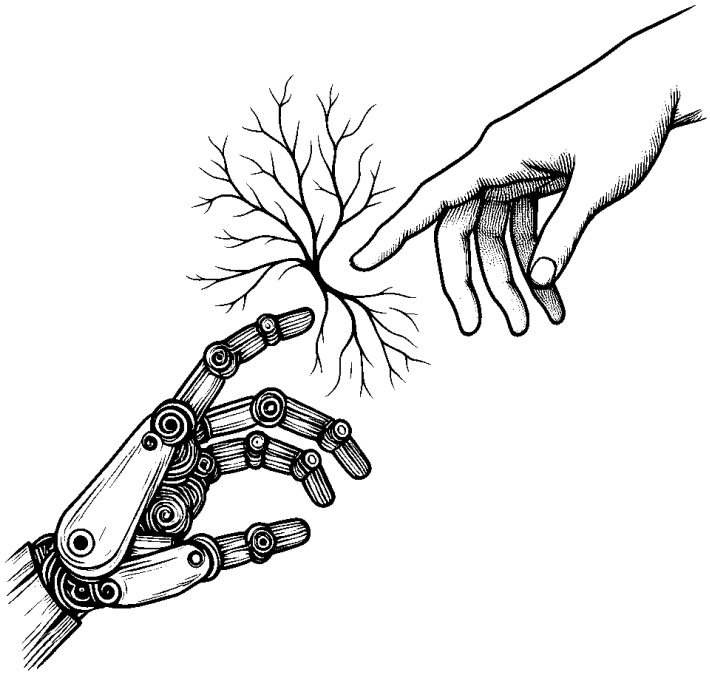
ReAIC:

Reactive Active Inference Control for Enhanced Performance of Robotic Manipulators

Alon Dawe

Master of Science Thesis



TU Delft Delft University of Technology

Department of Cognitive Robotics

ReAIC: Reactive Active Inference Control for Enhanced Performance of Robotic Manipulators

MASTER OF SCIENCE THESIS

For the degree of Master of Science in Robotics at
Delft University of Technology

Alon Dawe
Student Number: 5603250

April 9, 2024

To be defended publicly on Wednesday April 24, 2024

Supervisor:	Prof. Dr. R. Babuska
Committee Members:	Prof. Dr. Ir. M. Wisse
	Ing. G. Franzese
	Ir. C. van Hoof MBA

Faculty of Mechanical Engineering (ME) · Department of Cognitive Robotics (CoR) · Delft University of Technology

ReAIC: Reactive Active Inference Control for Enhanced Performance of Robotic Manipulators

Alon Dawe
Supervisor: Prof. Dr. R. Babuska
Delft University of Technology, Cognitive Robotics, MSc Robotics Thesis, April 9, 2024

***Abstract*— The application of the Free-Energy Principle (FEP) and Active Inference (AIF) in robotics is an emerging field, with prior implementations primarily focused on demonstrating its viability. While these prior works have shown great promise in functionality, they have often lacked sufficient attention to the tunability of these systems for addressing specific performance criteria. This paper introduces the ReAIC, a modified prior implementation of the Active Inference Controller (AIC). The modification enhances the controller's ability to influence performance criteria, such as Integral Time Absolute Error (ITAE), settling time and overshoot. A comprehensive tuning procedure is outlined alongside a detailed analysis revealing the performance impact of each tuning parameter. Comparative evaluations against three distinct controllers — the classic PID controller, a standard implementation of an AIC, and an Adaptive Friction Compensator (AFC) — are conducted across five experimental scenarios encompassing step response tests on a per-joint-basis of a physical 5-DOF robot arm. Results demonstrate the ReAIC's superior performance in gravity compensation tasks, exhibiting robust adaptability in the presence of dynamic environmental factors. Furthermore, comparative assessments reveal the ReAIC's sensitivity to induced friction, highlighting the AFC's superior performance under such conditions. This paper highlights the ReAIC's potential as a responsive and adaptable control framework, improving upon a previously implemented AIC.**

***Index Terms*— Free-Energy Principle, Active Inference Control, Adaptive Control of Robotic Systems.**

I. Introduction

For many years, robotic manipulation has been extensively used in industries to perform repetitive tasks in controlled environments. As robotic hardware and control software improve, and technological innovation continues to gain ground, there is an increased demand for robotic manipulators in extremely unstructured, uncertain and dynamic environments. For a manipulator to function effectively and autonomously with unknown inherent and external non-linear dynamic influences, the need for adaptive control algorithms and architectures that can cope with unexpected events or dynamic changes cannot be overstated.

A manipulator is better equipped to achieve specific criteria under uncertain and non-linear dynamics by utilising an adaptation mechanism that can adjust controller parameters online [1]. The field of adaptive control systems is an active research area that already includes numerous well-known and researched state-of-the-art adaptive control methods [2].

Recently, a unifying theory of brain function known as the *Free-Energy Principle* (FEP) has made its way from neuroscience to robotics and control [3]. The FEP encapsulates a mathematical expression detailing how biological organisms counteract the inherent inclination towards disorder in their environment by suggesting that they strive to minimise a single objective function known as *Variational Free Energy* (VFE) [4], [5]. The theory revolves around *Active Inference* (AIF), which describes the continual process of minimising VFE over time through deliberate actions [6]. This neuroscientific theory fundamentally suggests that cognition and motor control functions within the brain can be conceptualised with the minimisation of VFE.

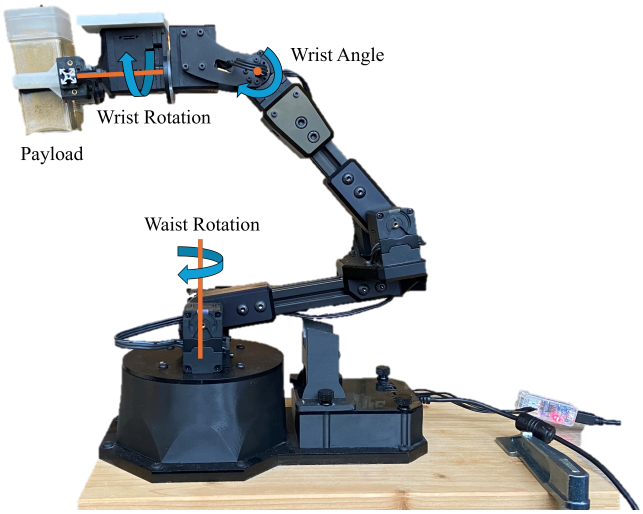


Fig. 1: Interbotix PincherX 150, 5-DOF Manipulator used in all experiments within this paper.

Employing AIF in robotic control holds promise for significant advancements in state estimation, adaptive control, planning, decision-making and learning, which form the foundation for autonomous robots [7]. This has already been shown to be the case in the few endeavours that utilise this AIF framework [8], the most relevant of which will be described in Section II. While these studies are all steps in the right direction and add value to AIF control in robotics, none except [9], attempt to present a tuning procedure or show how the tuning parameters influence the overall performance of the controller. In this paper, we modify a standard *Active Inference Controller* (AIC) [9] to enable tuning parameters to have more influence over specific performance criteria

and substantiate this with experiments on a real Interbotix PincherX 150 robotic manipulator as shown in Fig. 1. More specifically, our primary contributions are:

1) We present a modified active inference framework by adapting the dynamic generative reference model, building upon the AIC architecture introduced in [9]. Termed the *Reactive Active Inference Controller* (ReAIC), this modification is distinguished by its ability to be more reactive to reference command signals than the AIC.
2) The modification enables parameter tuning according to user-specific performance criteria, such as settling time and overshoot. Therefore, we offer a comprehensive tuning procedure alongside demonstrations of how the tuning parameters influence performance criteria.
3) We conduct a comparative analysis evaluating the adaptability and performance of the ReAIC relative to the AIC, PID and an *Adaptive Friction Compensator* (AFC) using an Interbotix PincherX 150, 5-DOF robotic manipulator. Notably, experiments focused on one degree of freedom at a time, as annotated in Fig. 1.

The sections of this paper are structured as follows: An overview of the related work is presented in Section II. Section III provides the derivation of the FEP, while Section IV uses this derivation and applies it to the ReAIC framework for a robotic manipulator. Section V details the controller tuning procedure and demonstrates how the tuning parameters influence certain performance criteria. Section VI provides experimental evaluations to demonstrate the performance of the ReAIC relative to the AIC, PID and AFC controllers. The results are analysed and discussed. Section VII discusses the advantages and limitations of the ReAIC. Section VIII concludes the paper.

II. Related Work

The first proof-of-concept case study to effectively utilise the FEP and AIF to control a robot was presented in [3]. A simulated 7-DOF arm of a PR2 robot was used in a series of target-reaching tests to shed light on the roles that proprioception and vision play within the framework by exposing these sensory modalities to noise. The control implementation was offline, in which the simulation did not directly use the control actions generated by the AIF framework and required an additional position controller to move the manipulator in both joint and cartesian space. No joint sensory feedback was provided from the simulation; instead, it was approximated using a dynamic model [10]. Despite this, the solution confirmed that active inference can be effectively used to control a robotic manipulator in simulation.

In another study [11], the first upper-body reaching and head object tracking was successfully implemented on a real iCub humanoid robot using the active inference framework. The solution used an online, closed-loop approach with velocity commands to test the robot's ability to reach for an object in 2D and 3D space using binocular vision when subjected to sensor noise and changes to the object's location. The approach proved computationally efficient, displaying online adaptation capabilities to changes in the object's visual location, and it was more accurate than an inverse kinematics approach.

These past implementations utilised knowledge of the true kinematics and dynamics of the manipulators to control the robot within cartesian space. In [9], a 7-DOF Franka Emika Panda manipulator was controlled in joint space using a modified active inference framework by defining a reference model instead of modelling the true dynamics of the manipulator. This proved to decrease computational complexity compared to previous approaches. The solution was an online, closed-loop approach using torque commands. Referred to as the AIC, its performance and adaptive properties were compared to a state-of-the-art *Model Reference Adaptive Controller* (MRAC). A series of pick-and-place manoeuvres with different payloads revealed that the AIC showed better adaptability performance and did not require any retuning from simulation to the real world, unlike the MRAC. The AIC was easier to tune due to the significantly fewer tuning parameters. While this AIC showed great promise, its implementation did not allow for tunability of desired performance criteria, such as rise time, settling time and overshoot.

Most drawbacks of the AIC in [9] were specifically highlighted in [12], which introduced a modified and extended *Unbiased Active Inference Control* (u-AIC) architecture. The term "unbiased" in the new architecture stems from a fault-tolerant control study [13], in which it was noted that the estimated manipulator joint states (state beliefs) are biased toward the goal. This discrepancy raises a fundamental question regarding the definition of state belief. We, the authors, emphasise that within the AIC architecture, the belief about the state does not represent the current state. Instead, it represents the desired reference state at which the current VFE is minimised while pursuing the goal state. Hence, it is crucial to distinguish between the reference state (belief) and the goal state within this paper, as they cannot be used interchangeably. The u-AIC architecture was applied to the same 7-DOF Franka Emika Panda manipulator and compared to the AIC in a reference tracking experiment. While the u-AIC outperformed the AIC in terms of reference tracking, this may have been due to the AIC's inability to be tuned for more responsiveness. Additionally, the u-AIC architecture only uses AIF as its state estimator, while a PID controller controls the manipulator.

III. Free-Energy Optimisation

This section covers the fundamental theoretical derivations and requirements for understanding and integrating AIF into robotics control. The descriptions, definitions and derivations of concepts presented in this section are primarily based on the writings of Buckley et al. [14] and Pezzato et al. [9], thanks to their mathematical relevance to control theory and robotics. An interested reader is referred to Appendix A for more detailed mathematical derivations and explanations of the expressions presented within this section.

A. Free-Energy Principle

The FEP is a promising candidate theory for a long-searched and much-anticipated unified brain theory in neuroscience. The FEP was initially introduced by Friston et al. [4], [5], who state that all (viable) biological organisms possess a tendency to resist disorder and therefore take actions to minimise *surprisal* in their habitable environment. Surprisal is a measure of the atypicality of an event or sensory input and can be quantified by the negative natural logarithm of the probability of the sensory data [14]:

$$-\ln p(y) \tag{1}$$

where $p(y)$ is the probability of observing some particular sensory data y in the organism's typical habitable environment.

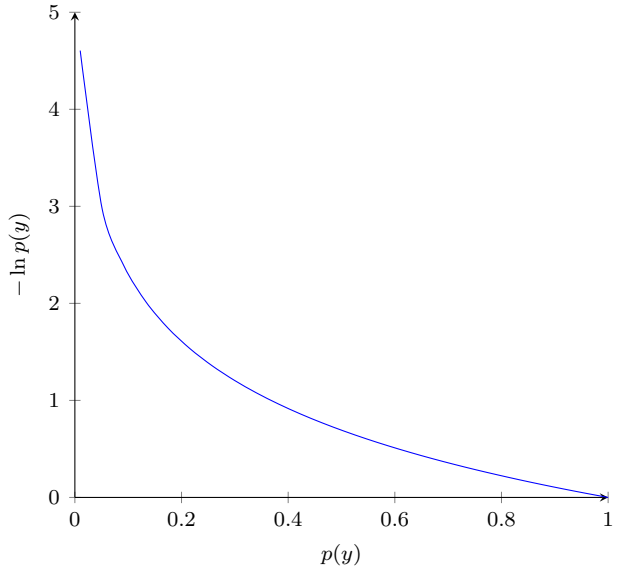


Fig. 2: Surprisal

Figure 2 illustrates that when the probability of an event is very low, organisms experience a heightened surprisal upon observing it, prompting them to take actions aimed at minimising this surprisal. The FEP argues, however, that the surprisal cannot be directly minimised [5]. Instead, the VFE is minimised by allowing an organism to act on its environment to change its sensory input.

Organisms maintain two central probabilistic models that can be defined in Bayesian terms using Bayes theorem:

$$p(x|y) = \frac{p(x,y)}{p(y)} = \frac{p(y|x)p(x)}{p(y)} = \frac{p(y|x)p(x)}{\int p(y|x)p(x)\,\mathrm{d}x} \tag{2}$$

The *posterior* $p(x|y)$ represents a probabilistic model of their typical habitable environment, which includes their own body. It achieves this by updating a multivariate Gaussian probability distribution over all possible states x, based on their observed sensory data y. This internal model of the environmental states is known as *recognition density* or *R-density* [5].

The more complex joint probability density $p(x, y)$ is the multiplication of the *likelihood* $p(y|x)$ and *prior* $p(y)$. It represents the probabilistic model of the sensory data and the environmental states occurring together. For R-density to be maintained or updated effectively, an organism must know how various environmental states x shape its sensory input y. In other words, if the organism needs to update its internal model of its environmental states $p(x|y)$, it needs to know which states cause certain sensory inputs. These implicit assumptions are known as *generative density* or *G-density* [5] and, in Bayesian terms, can be defined as the joint probability density between x and y as shown in the numerator of Eq. (2).

The integral in the denominator of Eq. (2) is often difficult or even practically impossible to calculate. To address this hurdle, a method known as *Variational Bayes* [15] is needed for determining the posterior $p(x|y)$ through an auxiliary probability distribution, which best resembles the posterior. This distribution is the R-density $r_d(x)$. Using the Kullback-Leibler divergence, a measure of the statistical difference between the R-density and the true posterior is formulated as [4]:

$$\begin{aligned} D_{KL}\left(r_d(x)||p(x|y)\right) &= \int r_d(x) \ln \frac{r_d(x)}{p(x|y)}\,\mathrm{d}x \\ &= \int r_d(x) \ln \frac{r_d(x)}{p(x,y)}\,\mathrm{d}x + \ln p(y) \\ &= \mathcal{F} + \ln p(y) \geq 0 \end{aligned} \tag{3}$$

where $\mathcal{F}$ is the VFE. It can be observed that minimising $\mathcal{F}$ by adjusting the R-density, such that $r_d(x) \approx p(x|y)$, allows for the Kullback-Leibler divergence between R-density and true posterior $D_{KL}(r_d(x)||p(x|y) = 0$. Since $D_{KL}(r_d(x) \geq 0$, this effectively sets $\mathcal{F}$ as an upper bound on the surprisal because it represents the maximum amount of free-energy that the organism can minimise to reduce its surprisal further [14].

B. R-Density: Encoding Environmental States

To minimise its VFE $\mathcal{F}$, an organism must explicitly encode its R-density $r_d(x)$. A standard approximation would be to assume that $r_d(x)$ is in the form of a Gaussian distribution, with mean μ and variance σ values of a single environmental state x [14].

$$r_d(x) \equiv \mathcal{N}(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma}\right\} \tag{4}$$

Under this approximation of the R-density, $\mathcal{F}$ can then be shown to approximate to (see Appendix A):

$$\mathcal{F} \approx -\ln p(\mu, y) \tag{5}$$

This equation suggests that an organism's internal representation of the environmental states x only considers the most likely environmental causes by using the mean μ of the R-density, neglecting the details of its distribution. In [5], the mean μ was called the believed state (belief).

C. G-Density: Encoding Environmental Causes

Until this point, how an organism encodes its G-density remains unclear – the relationship between environmental states x and their corresponding sensory data y. This is done by further specifying $p(\mu, y)$ from Eq. (5). Since $p(\mu, y) = p(y|\mu)p(\mu)$, the joint probability $p(\mu, y)$ can be defined by specifying two generative models.

1) Generative Model of the Sensory Data: A generative model of the sensory data y is generated by a non-linear mapping of the believed environmental states μ in combination with Gaussian noise $z \sim \mathcal{N}(0, \sigma_y)$ [14]:

$$y = g(\mu) + z \tag{6}$$

This equation represents the probabilistic likelihood $p(y|\mu)$, which can be expressed as:

$$p(y|\mu) = \frac{1}{\sqrt{2\pi\sigma_y}} \exp\left\{-\frac{(y - g(\mu))^2}{2\sigma_y}\right\} \tag{7}$$

2) Generative Model of the State Dynamics: A generative model that describes the dynamics of the belief μ, which would allow an organism to perform state inference on time-varying states, can be defined using a Langevin-type equation [16]:

$$\frac{\mathrm{d}\mu}{\mathrm{d}t} = \mu' = f(\mu) + w \tag{8}$$

where the function $f(\mu)$ specifies the desired evolution of the dynamics of the organism and w is Gaussian noise $w \sim \mathcal{N}(0, \sigma_\mu)$. Equation 8 represents the prior $p(\mu)$ which can be expressed as:

$$p(\mu) = \frac{1}{\sqrt{2\pi\sigma_\mu}} \exp\left\{-\frac{(\mu' - f(\mu))^2}{2\sigma_\mu}\right\} \tag{9}$$

D. Final Variational Free-Energy Expression

To find a final usable expression for the VFE $\mathcal{F}$, the *generalised motions* of the organism needs to be considered. Generalised motions are instantaneous temporal derivatives of the state variables, describing the evolution of the state of a dynamical system of an organism [15]. The generalised motions can be expressed as:

$$\begin{aligned} y &= y^{(0)} = g(\mu) + z & \mu' &= \mu^{(1)} = f(\mu) + w \\ y' &= y^{(1)} = \frac{\partial g}{\partial \mu}\mu' + z' & \mu'' &= \mu^{(2)} = \frac{\partial f}{\partial \mu}\mu' + w' \\ &\vdots & &\vdots \end{aligned} \tag{10}$$

where $z, z', \ldots$ and $w, w', \ldots$ are independent Gaussian noise sources from every dynamic order, and the prime symbol ($'$) indicates the time derivative $\frac{\mathrm{d}}{\mathrm{d}t}$.

Substituting Eqs. (7) and (9) into $\mathcal{F}$ while also considering the generalised motions up to an order n_d, the final VFE can be expressed as [14]:

$$\begin{aligned} \mathcal{F} = &\sum_{i=0}^{n_d-1}\left(\frac{1}{2\sigma_{y^{(i)}}}\epsilon_y^{(i)2} + \frac{1}{2}\ln\sigma_{y^{(i)}}\right) \\ &+ \sum_{i=0}^{n_d-1}\left(\frac{1}{2\sigma_{\mu^{(i)}}}\epsilon_\mu^{(i)2} + \frac{1}{2}\ln\sigma_{\mu^{(i)}}\right) \end{aligned} \tag{11}$$

where

$$\epsilon_y^{(i)} \equiv y^{(i)} - g^{(i)}(\mu) \qquad \epsilon_\mu^{(i)} \equiv \mu^{(i+1)} - f^{(i)}(\mu) \tag{12}$$

E. Perceptual and Active Inference

The process of minimising the VFE through perception involves an organism updating its internal belief about its habitable environment, R-density. It has been suggested that an organism's beliefs in its environmental states are updated according to a gradient descent scheme [14]. A gradient descent operation of dynamic systems, as shown in Eq. (13), also considers time derivatives through the block-matrix derivative operator D.

$$\Delta\mu^{(i)} = D\mu^{(i)} - \kappa_\mu \frac{\partial}{\partial \mu^{(i)}}\mathcal{F} \tag{13}$$

Here, κ_μ is the learning rate for the belief update. This gradient descent operation will change the belief, further minimising the VFE.

An organism can also act in its habitable environment to indirectly change its sensory input toward a desired state, thereby minimising its VFE and surprisal. For an organism to take action, it must know how its sensory input changes in the presence of actions.

Making the same suggestion as in the case of minimisation through perception, an organism is suggested to act on its environment in a gradient descent scheme as well [14]. Equation 14 shows how gradient descent is used to calculate the change in the action, which involves calculating the gradient of the VFE with respect to the control action update:

$$\begin{aligned} \Delta u &= -\kappa_a \frac{\partial}{\partial u}\mathcal{F} \\ &= -\kappa_a \frac{\partial y}{\partial u}\frac{\partial}{\partial y}\mathcal{F} \end{aligned} \tag{14}$$

where κ_a is the learning rate for the control action.

Equations 13 and 14 represent the final theoretical framework of the FEP and its minimisation using perception and active inference. With this information, the next section will present how this framework can be applied to robotic manipulators.

IV. Re-Active Inference Control

The derivation of a generic n-DOF manipulator controlled by the ReAIC framework will be described, as originally outlined in [9]. This section aims to provide the reader with the appropriate background knowledge to comprehend the modifications introduced within this paper.

Key variables introduced in Section III need to be re-defined specifically within the framework of a robotic manipulator for better understanding. Robotic manipulators are generally equipped with joint position and velocity sensors, and since manipulators can have multiple (n) degrees of freedom, a simplified scalar form can no longer be supported. In this paper the measured output is defined as $\boldsymbol{y} = \left[\boldsymbol{y}_q, \boldsymbol{y}_{\dot{q}}\right]^T \in \mathbb{R}^{2\times n}$. The goal state $\boldsymbol{\mu_g} \in \mathbb{R}^n$, is the target or final joint position the controller must achieve and maintain. The control action $\boldsymbol{u} \in \mathbb{R}^n$ is the command signal sent to the

manipulator's joints, responsible for changing the measured output to follow the reference state (belief) $\tilde{\boldsymbol{\mu}} \in \mathbb{R}^{n_d \times n}$. The reference state is generated by the perceptual inference operation Eq. (13). It encapsulates the instantaneous temporal state at which the manipulator's $\mathcal{F}$ is minimised and is represented by $\tilde{\boldsymbol{\mu}} = \begin{bmatrix} \boldsymbol{\mu}^{(0)} & \boldsymbol{\mu}^{(1)} & \dots & \boldsymbol{\mu}^{(n_d)} \end{bmatrix}^T$.

Using the theory presented in Section III, what remains is to define a suitable generative model of the sensory data $\boldsymbol{g}(\boldsymbol{\mu})$ and the dynamic generative model of the world $\boldsymbol{f}(\boldsymbol{\mu})$, as shown in Eqs. (6) and (8), respectively. Both equations depend on $\boldsymbol{\mu}$, representing the manipulator's internal reference joint position.

A. Generative Model of the Sensory Data

Since $\boldsymbol{g}(\boldsymbol{\mu})$ represents the relationship between the sensory position feedback $\boldsymbol{y}_q$ and the reference position $\boldsymbol{\mu}$, the generative model of the sensory data can be defined as:

$$\begin{aligned} \boldsymbol{y}_q &= \boldsymbol{g}(\boldsymbol{\mu}) + \boldsymbol{z} \\ &= \boldsymbol{\mu} + \boldsymbol{z} \end{aligned} \tag{15}$$

where $\boldsymbol{z}$ represents multidimentional Gaussian noise $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_y)$ and the covariance matrix $\boldsymbol{\Sigma}_y = \sigma_y \boldsymbol{I}_n$. σ_y represents the variance of the Gaussian noise defined in Eq. (6) of the univariate case, multiplied by a square identity matrix $\boldsymbol{I}_n$ of size n.

B. Dynamic Generative Model of the World

In [9], the authors proposed a novel idea to define a reference model as the dynamic generative model of the world. This reference model would specify the desired dynamic behaviour of the manipulator instead of modelling the true dynamics. This was done by defining $\boldsymbol{f}(\boldsymbol{\mu}) = \boldsymbol{\mu}_g - \boldsymbol{\mu}$. By doing this, the controller believes the joints will be steered towards the goal position $\boldsymbol{\mu}_g \in \mathbb{R}^n$ [9]. However, this definition of $\boldsymbol{f}(\boldsymbol{\mu})$ is incomplete when substituted into Eq. (8). The difference between the goal state $\boldsymbol{\mu}_g$ and the reference state $\boldsymbol{\mu}$ cannot represent the dynamics of a first-order system unless an assumption is made that the time constant $\tau = 1$. This would result in a hard-coded desired settling time of 5 s to a unit goal input $\mu_g = 1$ (99.3 % of μ_g at 5τ), as shown in the step responses presented in [9]. In this paper, we suggest a modification to this reference model to allow more control over the desired settling time by updating Eq. (8) to:

$$\begin{aligned} \boldsymbol{\mu}' &= \boldsymbol{f}(\boldsymbol{\mu}) + \boldsymbol{w} \\ &= \mathcal{K}_p(\boldsymbol{\mu}_g - \boldsymbol{\mu}) + \boldsymbol{w} \end{aligned} \tag{16}$$

where $\mathcal{K}_p$ is a tuning parameter that represents $1/\tau$. Decreasing the time constant τ (increasing $\mathcal{K}_p$) will imply a faster system response. This gives the controller a tunable sense of urgency and greater flexibility depending on the required performance criteria. $\boldsymbol{w}$ represents multidimensional Gaussian noise $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_\mu)$ and the covariance matrix $\boldsymbol{\Sigma}_\mu = \sigma_\mu \boldsymbol{I}_n$. σ_μ represents the variance of the Gaussian noise defined in Eq. (8) of the univariate case.

C. Perceptual and Active Inference of a Manipulator

With position and velocity sensory feedback, it was suggested [9] to define the generalised motions in Eq. (10) up to the second order $n_d = 2$. $\boldsymbol{g}^{(i)}(\boldsymbol{\mu})$ and $\boldsymbol{f}^{(i)}(\boldsymbol{\mu})$ as defined in Eq. (12) are:

$$\begin{aligned} \boldsymbol{g}^{(0)}(\boldsymbol{\mu}) &= \boldsymbol{\mu} & \boldsymbol{f}^{(0)}(\boldsymbol{\mu}) &= \mathcal{K}_p(\boldsymbol{\mu}_g - \boldsymbol{\mu}) \\ \boldsymbol{g}^{(1)}(\boldsymbol{\mu}) &= \boldsymbol{\mu}' & \boldsymbol{f}^{(1)}(\boldsymbol{\mu}) &= -\mathcal{K}_p \boldsymbol{\mu}' \end{aligned} \tag{17}$$

Substituting these equations into Eq. (11), the VFE of an n-DOF manipulator is defined as [9]:

$$\begin{aligned} \mathcal{F} = \quad & \frac{1}{2}(\boldsymbol{y}_q - \boldsymbol{\mu})^T \boldsymbol{\Sigma}_{y^{(0)}}^{-1} (\boldsymbol{y}_q - \boldsymbol{\mu}) \\ & + \frac{1}{2}(\boldsymbol{y}_{\dot{q}} - \boldsymbol{\mu}')^T \boldsymbol{\Sigma}_{y^{(1)}}^{-1} (\boldsymbol{y}_{\dot{q}} - \boldsymbol{\mu}') \\ & + \frac{1}{2}(\boldsymbol{\mu}' - \mathcal{K}_p(\boldsymbol{\mu}_g - \boldsymbol{\mu}))^T \boldsymbol{\Sigma}_{\mu^{(0)}}^{-1} (\boldsymbol{\mu}' - \mathcal{K}_p(\boldsymbol{\mu}_g - \boldsymbol{\mu})) \\ & + \frac{1}{2}(\boldsymbol{\mu}'' + \mathcal{K}_p \boldsymbol{\mu}')^T \boldsymbol{\Sigma}_{\mu^{(1)}}^{-1} (\boldsymbol{\mu}'' + \mathcal{K}_p \boldsymbol{\mu}') \end{aligned} \tag{18}$$

After establishing the VFE for the robotic manipulator, we implement gradient descent on $\mathcal{F}$ with respect to the reference state $\boldsymbol{\mu}^{(i)}$, as specified in Eq. (13) for each of the generalised motions up to the chosen order of n_d. This process yields [9]:

$$\begin{aligned} \Delta\boldsymbol{\mu} = \boldsymbol{\mu}' + \kappa_\mu \Big[& \boldsymbol{\Sigma}_{y^{(0)}}^{-1} (\boldsymbol{y}_q - \boldsymbol{\mu}) \\ & - \boldsymbol{\Sigma}_{\mu^{(0)}}^{-1} (\boldsymbol{\mu}' - \mathcal{K}_p(\boldsymbol{\mu}_g - \boldsymbol{\mu})) \Big] \end{aligned} \tag{19a}$$

$$\begin{aligned} \Delta\boldsymbol{\mu}' = \boldsymbol{\mu}'' + \kappa_\mu \Big[& \boldsymbol{\Sigma}_{y^{(1)}}^{-1} (\boldsymbol{y}_{\dot{q}} - \boldsymbol{\mu}') \\ & - \boldsymbol{\Sigma}_{\mu^{(0)}}^{-1} (\boldsymbol{\mu}' - \mathcal{K}_p(\boldsymbol{\mu}_g - \boldsymbol{\mu})) \\ & - \boldsymbol{\Sigma}_{\mu^{(1)}}^{-1} (\boldsymbol{\mu}'' + \mathcal{K}_p \boldsymbol{\mu}') \Big] \end{aligned} \tag{19b}$$

$$\Delta\boldsymbol{\mu}'' = -\kappa_\mu \boldsymbol{\Sigma}_{\mu^{(1)}}^{-1} (\boldsymbol{\mu}'' + \mathcal{K}_p \boldsymbol{\mu}') \tag{19c}$$

It is essential to mention that the reference state does not depict the current state but represents the desired state that would minimise $\mathcal{F}$ while pursuing the goal. In [17], this gradient descent operation was described as "hitting a moving target" because the reference dynamics "can be regarded as a gradient descent in a frame of reference that moves with the expected motion".

After deriving the new desired reference state in the generalised motions, the ReAIC utilises the same gradient descent scheme to generate control actions to guide the manipulator joints to the reference state. Equation 14 is simplified under the assumption that $\frac{\partial y}{\partial u}$ is represented by an identity matrix, implying that only the sign is considered in the relationship between sensory input y and control action u [9]. This assumption facilitates a more straightforward calculation of control actions:

$$\Delta\boldsymbol{u} = -\kappa_a \left[\boldsymbol{\Sigma}_{y^{(0)}}^{-1} (\boldsymbol{y}_q - \boldsymbol{\mu}) + \boldsymbol{\Sigma}_{y^{(1)}}^{-1} (\boldsymbol{y}_{\dot{q}} - \boldsymbol{\mu}') \right] \tag{20}$$

It is important to highlight that Eqs. (19a), (19b) and (19c) represent the change in the reference state and Eq. (20) represents the change in the control actions, aimed at minimising the VFE. Consequently, these equations are updated

via first-order Euler integration [9]. Algorithm 1 presents the pseudo-code for implementing the ReAIC in a 1-DOF scenario, where $n = 1$. The corresponding control diagram presented in Fig. 3 was adapted from [18].

D. State-Space Representation

Formulating the ReAIC in state-space representation provides a more flexible and compact framework for analyses and computer-aided design tools like MATLAB® [19]. All equations derived within this section can be expressed together in the state-variable form, which, for simplicity, has been illustrated for a single manipulator joint as [19]:

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}_c \tag{21}$$

$$y_c = \mathbf{C}\boldsymbol{x} \tag{22}$$

where the state vector $\boldsymbol{x}$ and controller input vector $\boldsymbol{u}_c$ are:

$$\boldsymbol{x} = \begin{bmatrix}\mu & \mu' & \mu'' & u\end{bmatrix}^T \quad \boldsymbol{u}_c = \begin{bmatrix}\mu_g & y_q & y_{\dot{q}}\end{bmatrix}^T$$

such that the system output y_c equates to the control action u of the ReAIC controller ($y_c = u$). The $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ matrices and the corresponding open-loop transfer function of the ReAIC have been derived in Appendix C.

V. ReAIC Tuning

The previous section introduced the ReAIC framework applied to an n-DOF manipulator. This section provides an in-depth overview of the tuning parameters and a detailed tuning procedure which builds on the tuning procedure of the AIC provided in [9]. Moreover, we highlight the influence of key tuning parameters on performance metrics, specifically the ITAE score, overshoot and settling time.

A. Tuning Parameters

The ReAIC control loop as shown in Algorithm 1, was designed to control one manipulator joint. It is easily expandable to multiple degrees of freedom as needed. The ReAIC requires seven tuning parameters per joint. This may seem like a lot, however, as will be explained in the tuning procedure outlined in Section V-B, only three of the seven would require delicate tuning. The tuning parameters are:

- $\sigma_q, \sigma_{\dot{q}}, \sigma_\mu, \sigma_{\mu'}$: Gaussian noise variances in the generative model of the sensory data $\sigma_q, \sigma_{\dot{q}}$ and the reference state dynamics $\sigma_\mu, \sigma_{\mu'}$. These variances represent the confidence in the controller's sensory position and velocity input and the confidence in the dynamic reference model velocity and acceleration, respectively [9].
- κ_μ, κ_a: The gradient descent learning rates for the reference state update and control actions, respectively. These are responsible for how fast the VFE is minimised through perception and action. Larger learning rates allow for a faster minimisation, although this can also cause gradient descent overshoot.
- $\mathcal{K}_p$: The proportional gain that distinguishes the ReAIC from the AIC defined in [9]. It influences the reference model speed at which the joints should be steered toward the goal position $\boldsymbol{\mu}_g$.

Algorithm 1 ReAIC Control

Initialisation:

 # Set tuning parameters and initial variable values

$\sigma_q, \sigma_{\dot{q}}, \sigma_\mu, \sigma_{\mu'}, \kappa_\mu, \kappa_a, \mathcal{K}_p$

$\Delta_t \leftarrow 0.001$

$u_{\max} \leftarrow 885$

$\kappa_{a\_\text{orig}} \leftarrow \kappa_a$

$u, u_{\text{prev}} \leftarrow 0$

 # Set μ, μ' to initial sensor values and μ'' to zero

$\mu \leftarrow y_q$

$\mu' \leftarrow y_{\dot{q}}$

$\mu'' \leftarrow 0$

$\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{\mu}}_{\text{prev}} \leftarrow \begin{bmatrix}\mu & \mu' & \mu''\end{bmatrix}^T$

Control Loop:

while true **do**

 # Retrieve position, velocity and goal state

$y_q, y_{\dot{q}}, \mu_g$

 # Joint angle within error tolerance?

if $|y_q - \mu_g| < 0.01$ **and** $y_{\dot{q}} = 0$ **then**

$\kappa_a \leftarrow 0$

else

$\kappa_a \leftarrow \kappa_{a\_\text{orig}}$

end if

 # Calculate reference state Eqs. (19)

$$\Delta\mu = \mu' + \kappa_\mu \left[\sigma_q{}^{-1}(y_q - \mu) - \sigma_\mu{}^{-1}(\mu' - \mathcal{K}_p(\mu_g - \mu))\right]$$

$$\Delta\mu' = \mu'' + \kappa_\mu \left[\sigma_{\dot{q}}{}^{-1}(y_{\dot{q}} - \mu') - \sigma_\mu{}^{-1}(\mu' - \mathcal{K}_p(\mu_g - \mu)) - \sigma_{\mu'}{}^{-1}(\mu'' + \mathcal{K}_p\mu')\right]$$

$$\Delta\mu'' = -\kappa_\mu \sigma_{\mu'}{}^{-1}(\mu'' + \mathcal{K}_p\mu')$$

$$\Delta\tilde{\boldsymbol{\mu}} = \begin{bmatrix}\Delta\mu & \Delta\mu' & \Delta\mu''\end{bmatrix}^T$$

 # Update reference state though integration

$\tilde{\boldsymbol{\mu}} \leftarrow \tilde{\boldsymbol{\mu}}_{\text{prev}} + \Delta_t \Delta\tilde{\boldsymbol{\mu}}$

$\tilde{\boldsymbol{\mu}}_{\text{prev}} \leftarrow \tilde{\boldsymbol{\mu}}$

 # Calculate control action Eq. (20)

$$\Delta u = -\kappa_a \left[\sigma_q{}^{-1}(y_q - \mu) + \sigma_{\dot{q}}{}^{-1}(y_{\dot{q}} - \mu')\right]$$

 # Update action through integration

$u \leftarrow u_{\text{prev}} + \Delta_t \Delta u$

 # Control action windup prevention

if $|u| > u_{\max}$ **then**

$u \leftarrow \text{sgn}(u)\, u_{\max}$

end if

$u_{\text{prev}} \leftarrow u$

execute_action(u)

end while

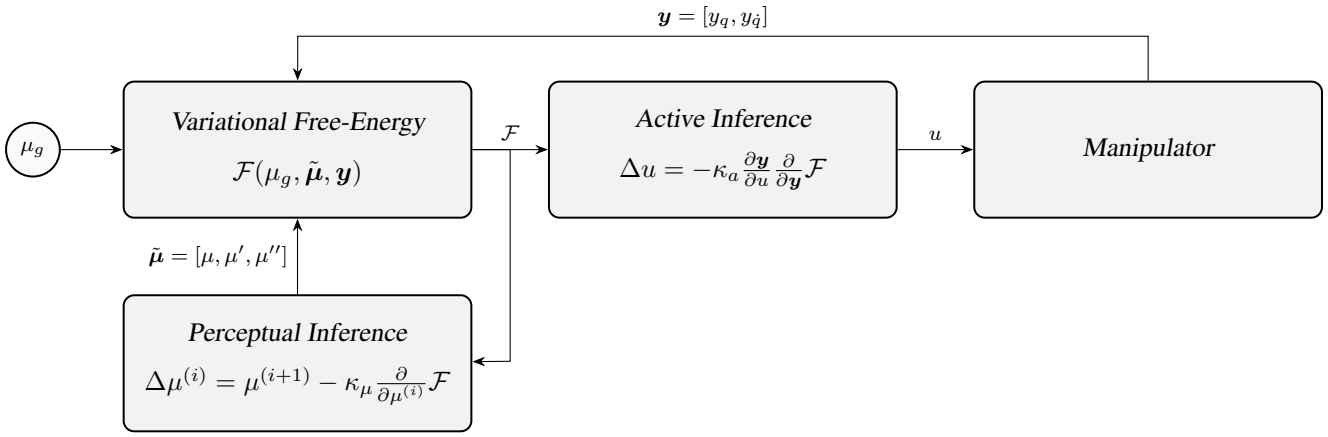


Fig. 3: ReAIC control diagram [18].

B. Tuning Procedure

The following tuning procedure was adapted from [9] as it proved to be the most convenient way to tune the parameters for a satisfactory response.

1) The variances $\sigma_q, \sigma_{\dot{q}}, \sigma_\mu, \sigma_{\mu'}$ and the proportional gain $\mathcal{K}_p$ are set to one, indicating high confidence in both the sensory input and reference state and setting a slow settling time.
2) The control actions are disabled by setting the learning rate κ_a to zero.
3) The learning rate for the reference state update κ_μ is incremented such that the state converges to a stable value in a static case within 20–30 ms [18]. This is done by plotting the reference state variable position μ during a step response while the control action has been disabled.
4) A desired step response settling time t_s can be set by increasing the proportional gain $\mathcal{K}_p$ with the following formula:
$$\mathcal{K}_p = \frac{5(\mu_g - y_q)}{t_s} \tag{23}$$
Increasing $\mathcal{K}_p$ to large values will result in overshoot behaviour in the real system (see Section V-C). It is suggested to start with a reasonable desired settling time.
5) Increment the control action's learning rate κ_a to allow the joint to be steered to the goal. A large learning rate is preferable for the controller to cope with changes in the sensory input as quickly as possible. However, increasing it too much will result in oscillatory behaviour due to overshooting the gradient descent operation.
6) If the system response continues to exhibit oscillatory behaviour despite increasing $\mathcal{K}_p$ and decreasing κ_a, adjusting the variances of the noisiest sensors and the reference state of the higher-order generalised motions becomes necessary to reduce their sensory confidence. When a large $\mathcal{K}_p$ is chosen, lowering the confidence in the higher-order generalised motion, denoted as $\sigma_{\mu'}$, becomes crucial. This is because the control actions will attempt to control the real manipulator joint (second-order system) using the first-order reference model at large velocities and accelerations. Therefore, the confidence in the reference model acceleration should be decreased.

C. Tuning Parameter Influence on Performance Metrics

The performance influences of the individual tuning parameters were tested on the single wrist rotation joint of a real Interbotix PincherX 150 robotic manipulator. [1]

Steps 1–3 of the tuning procedure were completed, leaving the control action learning rate κ_a and the proportional gain $\mathcal{K}_p$ influences to be analysed. It should be noted that by default, the variance $\sigma_{\mu'}$ was set to a large value due to the possibility of a large $\mathcal{K}_p$ causing unreliable reference velocities and accelerations for the controller, as mentioned in step 6.

Each combination of κ_a and $\mathcal{K}_p$ was tested on a series of seven goal step responses as depicted in Fig. 4. The *Integral Time Absolute Error* (ITAE) was computed as a measure of accuracy, and the settling time t_s and the overshoot M_p were observed for each step.

Figures 5(a), (b) and (c) represent the mean and Figs. 5(d), (e) and (f) represent the standard deviation (std) of each performance metric across all seven steps presented in 3D plots. Figures 5(a) and (b) reveal a valley formation with a steep gradient in the ITAE score and settling time when κ_a is small. This behaviour stems from the slow response of the control action, leading to delayed reactions to goal inputs or environmental changes. This is evident in the overshoot behaviour associated with small κ_a values and a large $\mathcal{K}_p$ as depicted in Fig. 5(c). As emphasised in step 5 of the tuning procedure, a larger κ_a is favoured.

[1]For more information on the Interbotix PincherX 150 robotic manipulator, please visit their website at https://docs.trossenrobotics.com/interbotix_xsarms_docs/specifications/px150.html

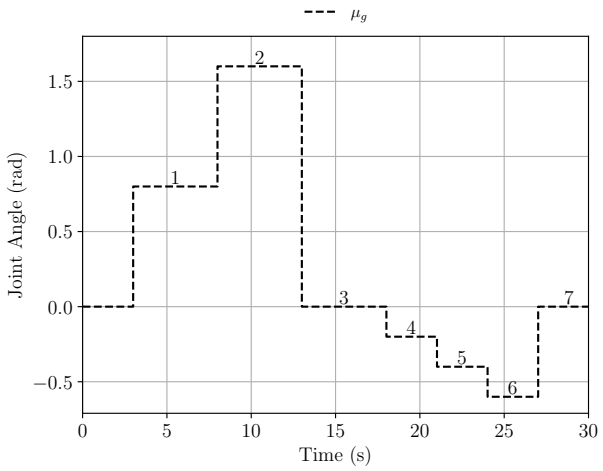


Fig. 4: Goal steps over time indicating each step for which ITAE, settling time and overshoot were observed.

Increasing κ_a while $\mathcal{K}_p$ is set to 1 provides an unsatisfactory ITAE score and a long settling time due to the oscillatory behaviour of the step response. The oscillatory behaviour is not noticeable within these plots, however, a slow settling time may suggest its presence. Increasing $\mathcal{K}_p$ with a larger κ_a improves the ITAE score and decreases the settling time, moving into the trough of the ITAE and settling time 3D plot valleys. Increasing $\mathcal{K}_p$ past the trough increases the overshoot, and as a result, the settling time and ITAE score increase.

Figures 5(d) and (e) reveal the same valley formations for each of the performance metrics. These 3D plots shed light on how the individual performance metrics vary according to the different step sizes shown in Fig. 4. Notably, selecting a combination of κ_a and $\mathcal{K}_p$ that yields optimal ITAE scores and minimal settling times results in the least deviation and greatest resilience to different step sizes. The corresponding response is illustrated in Fig. 6.

D. Tuning Parameter Influence on Sensor Noise

In the event of a noisy position and velocity sensor, the tuning parameters σ_μ and $\sigma_{\dot{\mu}}$ can be increased to decrease the confidence in the position and velocity feedback, respectively. Applying a Gaussian noise $\eta \sim \mathcal{N}(0, 0.1)$ to the position and velocity feedback of the wrist rotation joint, and using the same tuning as in the ideal response of Fig. 6, the step response is depicted in Fig. 7. Here, it can be observed that the noise causes undesirable fluctuations in the joint position. Since both position and velocity sensors are noisy, increasing the tuning parameters σ_μ and $\sigma_{\dot{\mu}}$ increases the assumed noise variance and thereby decreases the confidence in the sensory feedback. Figure 8 shows the influence of this, which results in a smoother response, with more overshoot the larger the assumed variance. Once satisfied with the smoothness of response, here $\sigma_\mu = \sigma_{\dot{\mu}} = 3$, the overshoot can be minimised by decreasing $\mathcal{K}_p$, as was explained in Section V-C. Changing $\mathcal{K}_p = 2$, with $\sigma_\mu = \sigma_{\dot{\mu}} = 3$, results in a much smoother response with less overshoot compared to the original noisy response.

VI. Experimental Evaluation

In this section, we systematically evaluate the ReAIC controller on a real 5-DOF Interbotix PincherX 150 robotic manipulator on a per-joint basis. Five different experiments were performed in which the ReAIC was compared to the original AIC [9] (Appendix E), a classic PID controller (Appendix D) and an AFC [20] (Appendix F). The analyses used the same goal steps as in Fig. 4, and their corresponding performance metrics – ITAE, settling time t_s and overshoot percentage M_p were measured. For each metric, the lowest value is considered to perform the best and is presented in **blue bold**, while the second-best performance metric is presented in **black bold**. The controller parameters were tuned for goal step 1 on the waist joint. No subsequent retuning was performed for any other tests. This specific manipulator was chosen due to the larger static friction inherent within each joint, compared to most other industry-leading manipulators, challenging the non-linear adaptation capabilities of each controller.

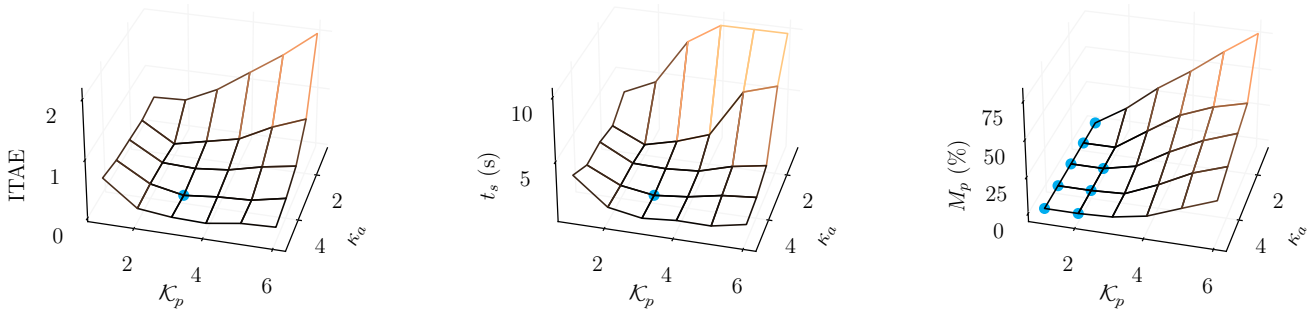
A. Experimental Setup and Parameters

Five experiments were performed on a real 5-DOF Interbotix PincherX 150 robotic manipulator. Two of these experiments focused on the waist joint and investigated how added friction affects the controller's performance. The other two experiments examined the effects of an added payload on the wrist angle joint. The last experiment was conducted on the wrist rotation joint and studied how the controllers cope with a change in the mass moment of inertia. See Fig. 12 for the physical setup of each experiment.

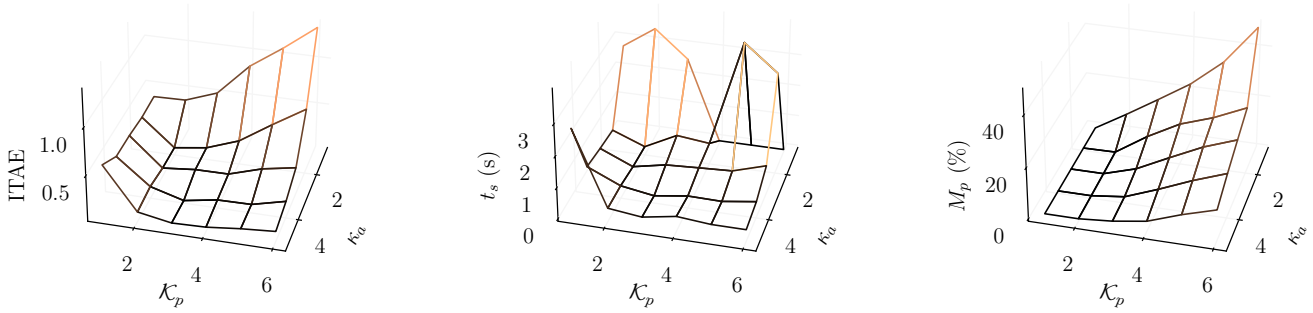
The manipulator's joints are actuated through DYNAMIXEL XL430-W250 smart servo motors, including a 12 bit, 360° contactless absolute position encoder. This allows for a resolution of 4096 steps per revolution, which results in 0.0015 rad per step. Each motor has a stall torque of 1.5 Nm (at 1.4 A) and a no-load speed of 61 rpm (at 12 V). Each motor was controlled through *Pulse Width Modulation* (PWM) commands ranging from $0 \sim 885$, where a command of 885 provides 12 V to the motor.

The tuning parameters of the ReAIC in all experiments were set to:

- $\sigma_q, \sigma_{\dot{q}}, \sigma_\mu = 1$: Variance representing the confidence of the sensory position and velocity feedback as well as the reference state.
- $\sigma_{\mu'} = 50$: Variance representing the confidence in the higher order reference state was set to a larger number due to the possibility of $\sigma_{\mu'}$ causing unwanted oscillations as described in step 6 of the tuning procedure outlined in Section V-B.
- $\kappa_\mu = 25.0$, $\kappa_a = 5.5$: The reference state and control actions learning rates respectively.
- $\mathcal{K}_p = 4.0$: The proportional gain was calculated using Eq. (23), for which the desired settling time was set to $t_s = 1$ for the goal step 1 $\mu_g = 0.8$.



(a) ITAE – mean (b) Settling Time – mean (c) Overshoot – mean



(d) ITAE – std (e) Settling Time – std (f) Overshoot – std

Fig. 5: Mean and std of performance metrics over all seven steps.

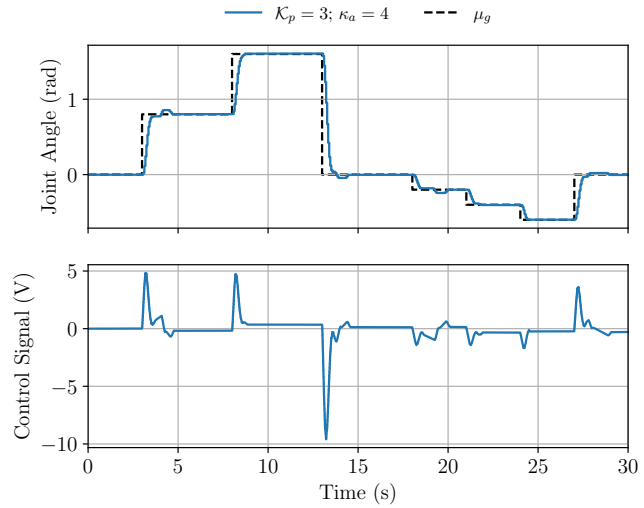


Fig. 6: ReAIC wrist rotation responses with optimal tuning parameters $\mathcal{K}_p = 3$ and $\kappa_a = 4$.

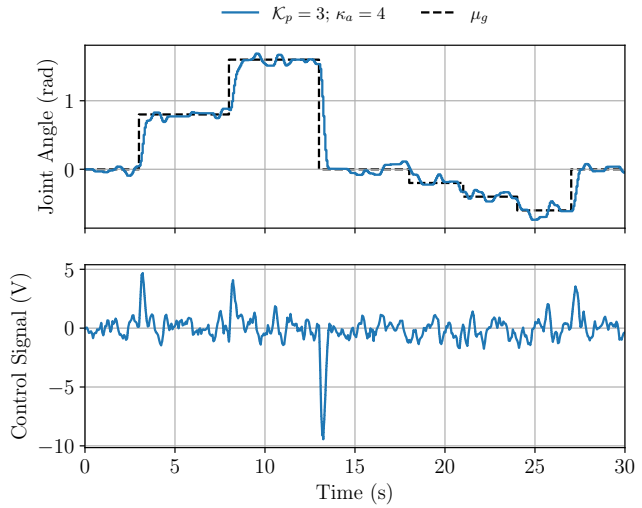


Fig. 7: ReAIC wrist rotation responses with induced Gaussian sensor noise and tuning parameters $\mathcal{K}_p = 3$ and $\kappa_a = 4$.

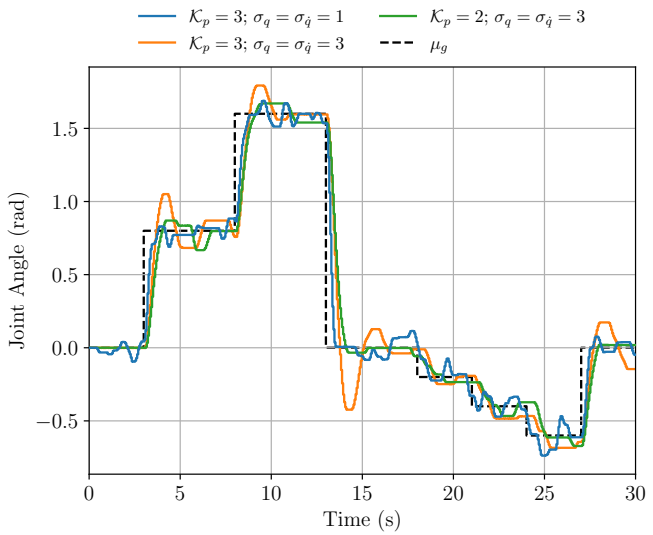


Fig. 8: ReAIC wrist rotation responses to increasing σ_q and $\sigma_{\dot{q}}$ and decreasing $\mathcal{K}_p$ with induced Gaussian sensor noise.

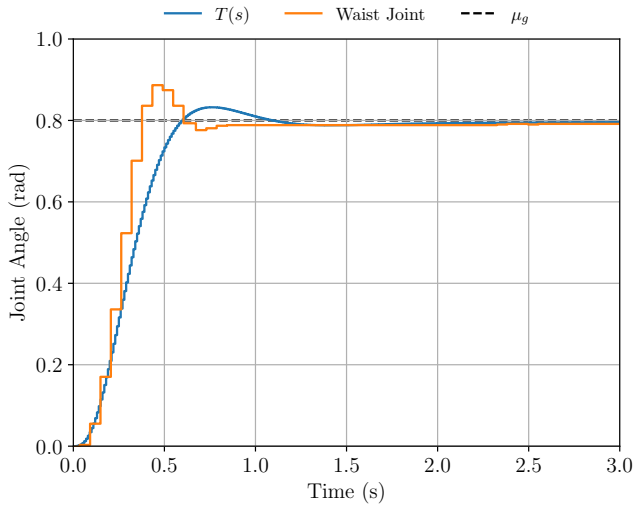


Fig. 9: ReAIC step response on the approximate closed-loop transfer function $T(s)$ in simulation and on the real manipulator waist joint.

The manipulator was controlled by all the controllers at a frequency of 1 kHz, with ROS [21] serving as the interface. All experiments were executed on a computer using Ubuntu 20.04 operating system with CPU: Intel® Core™ i7-10750H, GPU: NVIDIA® Quadro T1000, RAM: 32 GB. The source code for the experiments is freely available on GitHub. [2]

B. ReAIC Stability Analysis

An approximation of the manipulator joint transfer function is required to assess the stability of the closed-loop system. Each joint is assumed to be represented by a second-order dynamic model with the following transfer function:

$$G(s) = \frac{K_{\text{gain}}}{\tau s^2 + s} \tag{24}$$

The time constant τ and gain factor K_{gain} can be observed and calculated by performing an impulse response on the joint. Given that all controllers are tuned for the waist joint, an impulse response was conducted on this joint, yielding $\tau = 0.0934$ and $K_{\text{gain}} = 5.659$. See Appendix B for the detailed impulse response analysis.

Using the ReAIC state-space representation from Section IV-D, the simulated step response of the approximate closed-loop transfer function $T(s)$ is compared with the real manipulator waist joint's step response, as shown in Fig. 9. Given the close resemblance between the approximate transfer function $T(s)$ and the real system, examining the poles of the closed-loop transfer function will indicate the stability of the actual system. Figure 10 shows that the pole and zero locations of the closed-loop system with the ReAIC tuning parameters mentioned in Section VI-A are all in the left half-plane, indicating a stable system.

[2]For more information on the code implementation of all controllers, please visit: https://github.com/AlonDawe/interbotix_aic_control

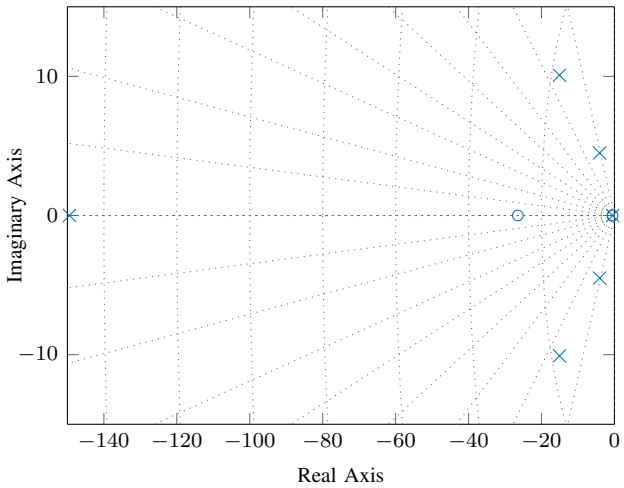


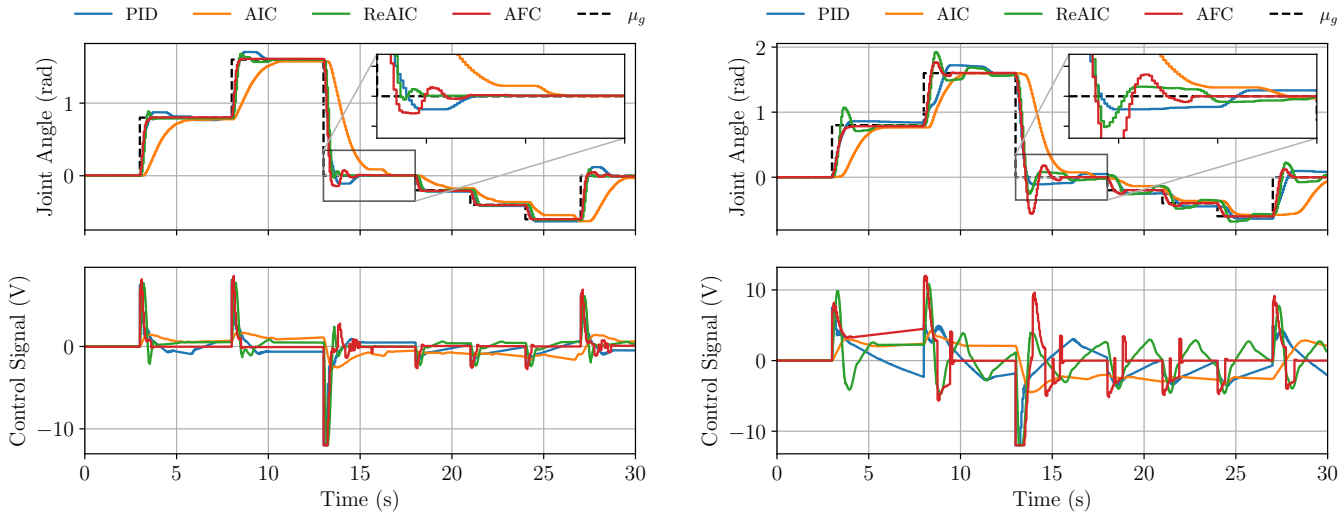
Fig. 10: Pole-Zero map of the closed-loop system.

$$\text{Poles} = \begin{bmatrix} -149.4 & -15.1 \pm 10.1\,\text{i} & -4.1 \pm 4.5\,\text{i} & -0.5 \end{bmatrix}^T$$
$$\text{Zeros} = \begin{bmatrix} -26.5 & -0.5 \end{bmatrix}^T$$

It is important to note that this stability analysis may only hold for the approximate linear model, which might not capture the real system's nuances or nonlinearities under different circumstances. Further proof of stability is beyond the scope of this paper.

C. Initial Controller Evaluation

The initial evaluation of each controller was performed on the manipulator waist joint. This joint operates in the horizontal plane, so gravity has no influence. The controller parameters were tuned for a settling time of $t_s = 1$ s, with an overshoot not exceeding 10 % on the first step. Additionally, to give all controllers a fair start, the maximum control signal of each controller was to be almost the same for the first goal



(a) Initial controller evaluation

(b) Additional friction

Fig. 11: Controller responses on the waist joint for (a): the initial controller evaluation and (b): with additional friction.

TABLE I: Performance metric results from the initial controller evaluation on the waist joint across all seven goal steps. The best performance is presented in **blue bold** and the second-best in **black bold**.

		PID	AIC	ReAIC	AFC
ITAE	mean	$1.80{\cdot}10^{-1}$	$7.01{\cdot}10^{-1}$	$\mathbf{9.00{\cdot}10^{-2}}$	$\mathbf{7.40{\cdot}10^{-2}}$
	std	$9.13{\cdot}10^{-2}$	$4.84{\cdot}10^{-1}$	$\mathbf{4.42{\cdot}10^{-2}}$	$\mathbf{5.66{\cdot}10^{-2}}$
t_s (s)	mean	$1.53{\cdot}10^{0}$	$2.87{\cdot}10^{0}$	$\mathbf{9.25{\cdot}10^{-1}}$	$\mathbf{8.27{\cdot}10^{-1}}$
	std	$6.24{\cdot}10^{-1}$	$1.07{\cdot}10^{0}$	$\mathbf{1.50{\cdot}10^{-1}}$	$\mathbf{3.55{\cdot}10^{-1}}$
M_p (%)	mean	$8.08{\cdot}10^{0}$	$\mathbf{0.00{\cdot}10^{0}}$	$5.52{\cdot}10^{0}$	$\mathbf{3.57{\cdot}10^{0}}$
	std	$\mathbf{2.77{\cdot}10^{0}}$	$\mathbf{0.00{\cdot}10^{0}}$	$3.11{\cdot}10^{0}$	$4.67{\cdot}10^{0}$

TABLE II: Performance metric results from the additional friction test on the waist joint across all seven goal steps. The best performance is presented in **blue bold** and the second-best in **black bold**.

		PID	AIC	ReAIC	AFC
ITAE	mean	$5.21{\cdot}10^{-1}$	$8.90{\cdot}10^{-1}$	$\mathbf{4.20{\cdot}10^{-1}}$	$\mathbf{1.43{\cdot}10^{-1}}$
	std	$2.76{\cdot}10^{-1}$	$7.50{\cdot}10^{-1}$	$\mathbf{2.15{\cdot}10^{-1}}$	$\mathbf{1.67{\cdot}10^{-1}}$
t_s (s)	mean	$3.07{\cdot}10^{0}$	$2.94{\cdot}10^{0}$	$\mathbf{2.84{\cdot}10^{0}}$	$\mathbf{1.35{\cdot}10^{0}}$
	std	$1.31{\cdot}10^{0}$	$1.39{\cdot}10^{0}$	$\mathbf{6.21{\cdot}10^{-1}}$	$\mathbf{6.44{\cdot}10^{-1}}$
M_p (%)	mean	$\mathbf{1.08{\cdot}10^{1}}$	$\mathbf{0.00{\cdot}10^{0}}$	$2.41{\cdot}10^{1}$	$1.70{\cdot}10^{1}$
	std	$7.69{\cdot}10^{0}$	$\mathbf{0.00{\cdot}10^{0}}$	$\mathbf{6.52{\cdot}10^{0}}$	$1.83{\cdot}10^{1}$

step, except for the AIC, for which it was impossible to tune for more responsiveness.

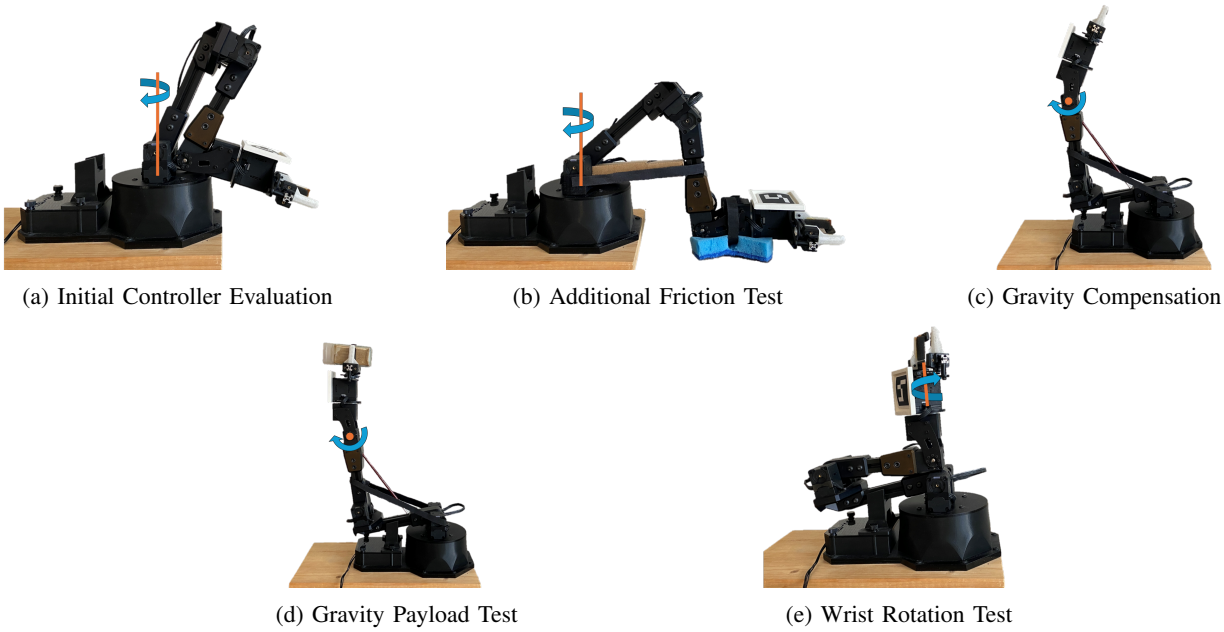
1) Qualitative results: The controller responses from the waist rotation step response test are shown in Fig. 11(a). The AIC has a noticeably slower response compared to the other controllers. The PID, ReAIC and AFC exhibit larger overshoot behaviour for the bigger goal steps 1–3 and 7 and negligible overshoot for the smaller goal steps 4–6. Additionally, intermittent motion or stick-slip behaviour can be observed when the control signal is non-zero due to the inherent static friction within the joint.

2) Quantitative results: The results of the controller mean and standard deviation of performance metrics over all seven steps are shown in Tab. I. The AFC outperformed the other controllers by achieving the lowest mean ITAE score and the fastest mean settling time, with the ReAIC closely following in second place. The ReAIC had the most consistent ITAE and settling time performance over all seven steps by achieving the lowest standard deviations, with the AFC landing in second place. The AIC's slower responsiveness enables it to achieve zero overshoot entirely, while the AFC achieves the second least amount of mean overshoot. The PID controller was the second most consistent in its overshoot behaviour, however it had the largest mean overshoot.

D. Additional Friction

This test investigated the impact of introducing friction to the waist joint on controller response. Static and dynamic friction of the waist joint was added by strapping a sponge to the gripper and performing the goal steps while sliding the sponge over the surface of a table. See Fig. 12(b) for the physical setup.

1) Qualitative results: The controller responses from the waist rotation test with additional friction are shown in Fig. 11(b). The obvious observation is the increased overshoot and stick-slip behaviour in all controllers. The AFC seems to cope extremely well with the added friction and manages to steer the joint towards the goal position faster than all other controllers. The AFC exhibits a much larger overshoot in the largest goal step 3, while the PID and ReAIC seem to be the only controllers with consistent overshoot performance across all steps. Another notable observation is that the ReAIC exhibits substantial seeking behaviour. Upon reaching a control signal of sufficient magnitude to



(a) Initial Controller Evaluation (b) Additional Friction Test (c) Gravity Compensation Test

(d) Gravity Payload Test (e) Wrist Rotation Test

Fig. 12: Physical manipulator setup for each test.

surpass static friction, the joint overshoots the goal position, unlike the AFC. The AIC has a slightly more delayed response compared to the initial controller evaluation due to the additional time needed to converge to a control signal large enough to overcome the additional friction of the joint. Contrary to the continued control transition of the other controllers, the AFC exhibits extremely jerky behaviour. This is due to the control algorithm trying to cope with the additional friction.

2) Quantitative results: It was expected and confirmed in Tab. II that the AFC would outperform all other controllers regarding overall ITAE performance, while the ReAIC achieved second-best. The AFC had the fastest mean settling time, followed by the ReAIC. However, the ReAIC had the most consistent settling time across all steps by having the lowest standard deviation. As expected, the AIC had the lowest overshoot behaviour, yet the PID had the second-lowest mean overshoot. The ReAIC had the second most consistent overshoot behaviour across all steps.

E. Gravity Compensation

In this test, the controllers' ability to handle gravitational influence on the manipulator's wrist angle rotation was assessed. The manipulator joint initiates from a vertically aligned position, corresponding to a joint orientation of zero radians. The AFC derived in [20] cannot compensate for the forces of gravity. For all tests that require gravity compensation, a modification of the AFC control algorithm was implemented to include an integrator within its control structure. The details of the changes are described in Appendix F.

1) Qualitative results: The individual controller responses for the gravity compensation test are shown in Fig. 13(a). Here, all controllers can reach the goal position with ease. The AIC shows a lot of stick-slip behaviour during the small goal steps 4–6, possibly due to friction or the controller attempting to counteract the forces of gravity. Apart from the AIC, the ReAIC consistently displays much less overshoot than the PID and AFC controllers.

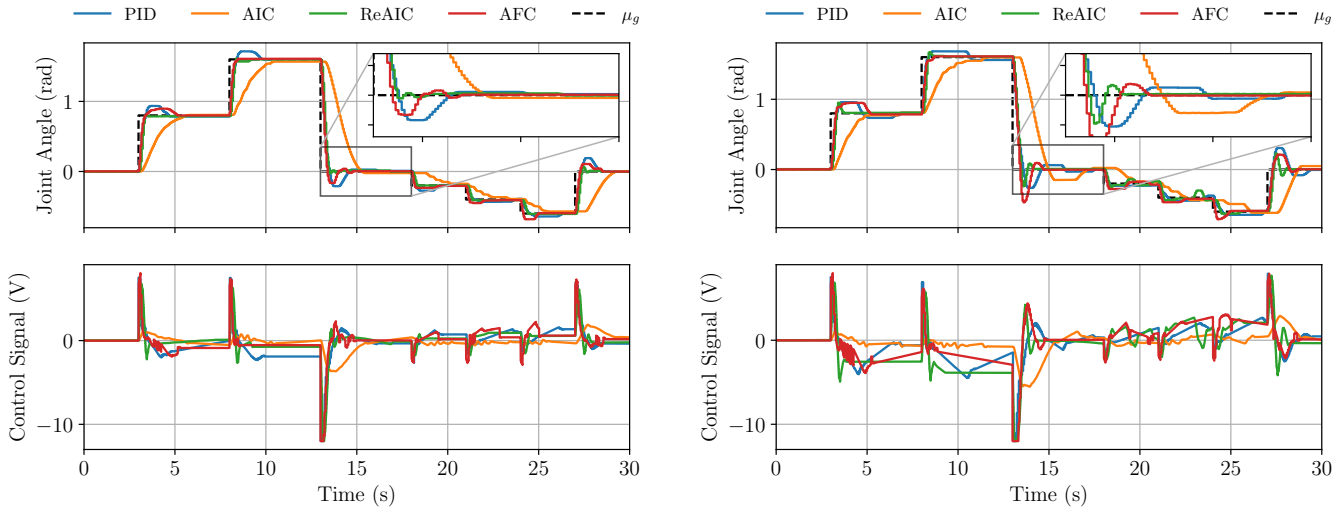
2) Quantitative results: The quantitative results of the gravity compensation test are shown in Tab. III. The ReAIC outperformed all other controllers in terms of the mean ITAE, with the AFC achieving second-best. The AFC showed the most consistent ITAE results across all steps. The ReAIC had the most consistent and fastest settling times compared to all other controllers. Again, the AIC had the least overshoot, followed by the ReAIC.

F. Additional Payload

This test evaluated the controllers' ability to cope with payload changes by adding a 35 g mass to the wrist angle, thereby increasing the weight influence on the joint. The modified AFC algorithm was used to cope with the gravitational forces.

1) Qualitative results: Figure 13(b) shows the controller responses for the payload test. A clear observation of the increased overshoot by all controllers, including the AIC, can be seen in the largest goal step 3 and the last step 7. This is due to the joint position of zero radians being a vertically aligned position in which the influence of gravity is at its lowest. All controllers seem to cope fairly well with the added payload by successfully managing to direct the joints to the goal position. All controllers seem to exhibit longer stick-slip intervals, probably because it takes longer for the control signal to account for the added weight influence.

2) Quantitative results: Table IV shows that the ReAIC had the best overall performance in terms of the mean and standard deviation of ITAE and settling time, with the AFC



(a) Gravity compensation

(b) Additional payload

Fig. 13: Controller responses on the wrist angle joint for (a): the gravity compensation test and (b): with an additional 35 g payload.

TABLE III: Performance metric results from the gravity compensation test on the wrist angle joint across all seven goal steps. The best performance is presented in **blue bold** and the second-best in **black bold**.

		PID	AIC	ReAIC	AFC
ITAE	mean	$1.83{\cdot}10^{-1}$	$5.85{\cdot}10^{-1}$	$\mathbf{7.90{\cdot}10^{-2}}$	$\mathbf{9.69{\cdot}10^{-2}}$
	std	$1.05{\cdot}10^{-1}$	$4.24{\cdot}10^{-1}$	$\mathbf{6.17{\cdot}10^{-2}}$	$\mathbf{4.37{\cdot}10^{-2}}$
t_s (s)	mean	$1.57{\cdot}10^{0}$	$2.21{\cdot}10^{0}$	$\mathbf{7.95{\cdot}10^{-1}}$	$\mathbf{1.35{\cdot}10^{0}}$
	std	$4.89{\cdot}10^{-1}$	$\mathbf{4.42{\cdot}10^{-1}}$	$\mathbf{1.76{\cdot}10^{-1}}$	$6.22{\cdot}10^{-1}$
M_p (%)	mean	$1.69{\cdot}10^{1}$	$\mathbf{5.23{\cdot}10^{-1}}$	$\mathbf{2.72{\cdot}10^{0}}$	$1.21{\cdot}10^{1}$
	std	$1.02{\cdot}10^{1}$	$\mathbf{8.09{\cdot}10^{-1}}$	$\mathbf{1.92{\cdot}10^{0}}$	$5.15{\cdot}10^{0}$

TABLE IV: Performance metric results from the added payload test on the wrist angle joint across all seven goal steps. The best performance is presented in **blue bold** and the second-best in **black bold**.

		PID	AIC	ReAIC	AFC
ITAE	mean	$3.63{\cdot}10^{-1}$	$6.88{\cdot}10^{-1}$	$\mathbf{1.44{\cdot}10^{-1}}$	$\mathbf{1.96{\cdot}10^{-1}}$
	std	$2.03{\cdot}10^{-1}$	$5.77{\cdot}10^{-1}$	$\mathbf{5.36{\cdot}10^{-2}}$	$\mathbf{1.21{\cdot}10^{-1}}$
t_s (s)	mean	$2.16{\cdot}10^{0}$	$2.65{\cdot}10^{0}$	$\mathbf{9.55{\cdot}10^{-1}}$	$\mathbf{1.58{\cdot}10^{0}}$
	std	$1.18{\cdot}10^{0}$	$8.85{\cdot}10^{-1}$	$\mathbf{2.60{\cdot}10^{-1}}$	$\mathbf{5.65{\cdot}10^{-1}}$
M_p (%)	mean	$1.68{\cdot}10^{1}$	$\mathbf{2.85{\cdot}10^{0}}$	$\mathbf{1.56{\cdot}10^{1}}$	$2.30{\cdot}10^{1}$
	std	$8.83{\cdot}10^{0}$	$\mathbf{5.25{\cdot}10^{0}}$	$\mathbf{8.28{\cdot}10^{0}}$	$1.41{\cdot}10^{1}$

achieving the second-best performance. The best overshoot performance was achieved by the AIC, with the ReAIC coming in second place.

G. Wrist Rotation

In this setup, the manipulator's wrist rotation joint was controlled horizontally to evaluate the controllers' performance during wrist rotation. This test is crucial as it reveals how well the controllers can handle a significant decrease in the mass moment of inertia dynamics. The wrist rotation joint, being characterised by the least mass moment of inertia, offers insights into the controllers' capabilities in such scenarios.

1) Qualitative results: The responses of the wrist rotation test depicted in Fig. 14 show a similar response to the initial controller evaluation test in Fig. 11(a). The PID controller shows consistently larger overshoot behaviour across all steps compared to the other controllers and the initial controller evaluation. This is probably due to the controller being tuned on the larger mass moment of inertia of the waist joint; as a result, the decrease in the mass moment of inertia causes more overshoot. The AFC shows less underdamped oscillations on the largest goal step 3, compared to its performance in Fig. 11(a).

2) Quantitative results: The final quantitative results of the wrist rotation test are presented in Tab. V. The AFC achieved the best mean ITAE and settling time performance, followed by the ReAIC. However, the ReAIC outperformed the AFC regarding settling time consistency and overall overshoot performance across all steps. The AIC had the least overshoot.

TABLE V: Performance metric results from the wrist rotation test across all seven steps. The best performance is presented in **blue bold** and the second-best in **black bold**.

		PID	AIC	ReAIC	AFC
ITAE	mean	$1.97{\cdot}10^{-1}$	$6.86{\cdot}10^{-1}$	$\mathbf{9.11{\cdot}10^{-2}}$	$\mathbf{5.88{\cdot}10^{-2}}$
	std	$1.02{\cdot}10^{-1}$	$5.17{\cdot}10^{-1}$	$\mathbf{5.32{\cdot}10^{-2}}$	$\mathbf{3.72{\cdot}10^{-2}}$
t_s (s)	mean	$1.44{\cdot}10^{0}$	$2.32{\cdot}10^{0}$	$\mathbf{8.42{\cdot}10^{-1}}$	$\mathbf{7.69{\cdot}10^{-1}}$
	std	$4.81{\cdot}10^{-1}$	$6.89{\cdot}10^{-1}$	$\mathbf{2.13{\cdot}10^{-1}}$	$\mathbf{4.05{\cdot}10^{-1}}$
M_p (%)	mean	$1.29{\cdot}10^{1}$	$\mathbf{0.00{\cdot}10^{0}}$	$\mathbf{4.21{\cdot}10^{-1}}$	$9.86{\cdot}10^{-1}$
	std	$1.08{\cdot}10^{1}$	$\mathbf{0.00{\cdot}10^{0}}$	$\mathbf{5.80{\cdot}10^{-1}}$	$2.12{\cdot}10^{0}$

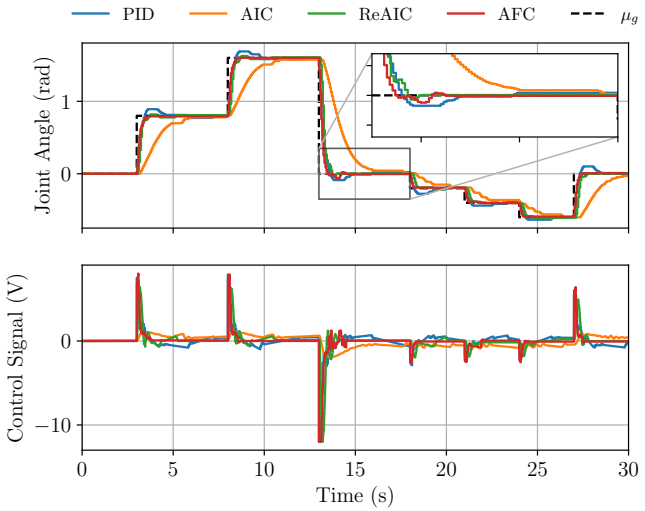


Fig. 14: Controller responses on the wrist rotation joint.

H. Discussion of Experiments

Comparing the percentage change in performance metrics between two tests and their absolute performance within each test can give insights into a controller's sensitivity, robustness and adaptability in changing system dynamics. By comparing the performance metrics of the initial controller evaluation with both the additional friction test and the wrist rotation test, we can evaluate each controller on its ability to cope with additional friction and changing mass moment of inertia. Additionally, comparing the performance metrics of the gravity compensation test with the added payload test provides insight into each controller's ability to cope with different payloads.

The lowest percentage changes are presented in **blue bold**, the second-lowest change is presented in **black bold** and the largest percentage change is presented in **orange bold**. Note that a positive percentage change indicates a performance degradation or decline, while a negative change indicates a performance improvement.

1) Initial controller evaluation vs. Additional friction: Table VI shows the percentage change in the performance metrics of the waist rotation joint between the initial controller evaluation and the additional friction test.

TABLE VI: Comparison of initial controller evaluation with the additional friction test on the waist joint. **Blue bold** indicates the smallest performance percentage change, followed by **black bold**, while **orange bold** indicates the largest.

		PID	AIC	ReAIC	AFC
Δ ITAE (%)	mean	189.36	**26.97**	**366.53**	**92.99**
	std	202.76	**55.11**	**387.01**	**195.07**
Δ t_s (%)	mean	100.93	**2.39**	**207.33**	**63.48**
	std	110.34	**30.22**	**314.00**	**81.67**
Δ M_p (%)	mean	**33.49**	**0.00**	336.12	**376.05**
	std	177.16	**0.00**	**109.83**	**291.27**

The ReAIC exhibited the largest ITAE and settling time changes in the presence of additional friction (largest performance decline). However, its absolute performance remained the same, achieving the second-best performance of all controllers in both tests. This may suggest that the controller is highly sensitive to changes in system dynamics induced by friction. Yet despite its sensitivity to friction, it maintains a relatively good performance compared to the other controllers in both tests. This suggests that while the controller's performance degrades with the introduction of friction, it remains robust enough to achieve satisfactory results.

The AIC had the lowest percentage changes in the performance metrics across both tests (smallest performance decline), and its absolute performance in terms of ITAE and settling time can be considered the worst relative to all other controllers. While this might suggest minimal sensitivity to additional friction, it is difficult to deduce whether this is a display of robustness and adaptability, given that it consistently performs poorly relative to the other controllers. Since the AIC is essentially a ReAIC controller with the proportional gain parameter $\mathcal{K}_p = 1$, this could imply that decreasing $\mathcal{K}_p$ entails a trade-off, where performance is traded for less sensitivity to additional friction.

The AFC achieved the second-lowest changes in ITAE and settling time (slight performance decline), while consistently outperforming all controllers regarding absolute ITAE and settling time performance in both tests. Therefore, despite slight changes in performance, the controller's sensitivity to introducing friction is relatively low compared to other controllers, displaying effective friction compensation. The controller exhibits the most robust and adaptive behaviour in its performance, as evidenced by its ability to maintain excellent absolute ITAE and settling time performance across both tests. It also achieved the largest percentage change in overshoot (largest performance decline), moving its ranking in absolute overshoot performance from second to third. This suggests that while the AFC can adapt well to additional friction in terms of ITAE and settling time, this comes at the expense of more overshoot behaviour.

The PID controller had the second-largest percentage changes in ITAE and settling time performance (large performance decline) and the second-lowest percentage change in overshoot (slight performance decline). While its absolute overall performance achieved third-best in each test, this indicates that the PID is very sensitive to additional friction, causing large performance degradations.

2) Initial controller evaluation vs. Wrist rotation test: The percentage change in the performance metrics of the initial controller evaluation and the wrist rotation test is shown in Tab. VII.

In this comparison, the AFC achieved the largest ITAE percentage changes (largest performance improvement). It also displayed the second-lowest settling time percentage changes (slight performance improvement), but the second-largest overshoot percentage change (large performance improvement). This might suggest that the AFC is sensitive

TABLE VII: Comparison of initial controller evaluation on the waist joint with the wrist rotation. **Blue bold** indicates the smallest performance percentage change, followed by **black bold**, while **orange bold** indicates the largest.

		PID	AIC	ReAIC	AFC
Δ ITAE (%)	mean	9.15	**-2.17**	**1.32**	**-20.57**
	std	**11.78**	**6.86**	20.20	**-34.29**
Δ t_s (%)	mean	**-5.54**	**-19.38**	-8.90	**-6.99**
	std	**-22.82**	-35.63	**41.90**	**14.11**
Δ M_p (%)	mean	**59.87**	**0.00**	**-92.38**	-72.40
	std	**288.68**	**0.00**	-81.35	**-54.66**

to the mass moment of inertia dynamic changes and that an increase thereof may negatively influence its ITAE and overshoot performance, but has less influence on its settling time. However, it remains robust and adaptive enough to achieve the best relative ITAE and settling time results in the presence of dynamic inertial changes.

The ReAIC had the most consistent ITAE performance across both tests, achieving the smallest percentage change (small performance decline), the second-largest change in settling time (large performance improvement) and the largest percentage change in overshoot (largest performance improvement). This could suggest that an increase in mass moment of inertia may negatively affect the overshoot and settling time performance but also indicate consistency of the ITAE performance when faced with these inertial changes. The ReAIC consistently achieved second-best absolute ITAE and settling time performance in both tests, exhibiting robust adaptability in the presence of changing dynamic factors.

The AIC showed consistent ITAE performance by achieving the second-lowest ITAE percentage changes (small performance improvement) and the largest percentage change in settling time (largest performance improvement). In the previous comparison, it was deduced that the AIC is the least sensitive to additional fiction. Therefore, inherent frictional characteristics in each joint can be disregarded as a cause. The lower mass moment of inertia may have required less control torque from the controller, allowing the wrist rotation to move toward its goal position much faster. Given that the AIC is a non-aggressive controller and that the absolute ITAE and settling time performance of both tests were consistently the worst, one might suggest that the only performance metric directly sensitive to a change in the joint inertial dynamics is its settling time, which indirectly influences its ITAE performance.

The PID controller had the second-largest percentage change in ITAE (large performance decline), with the smallest percentage change in settling time (small performance improvement) and the second-smallest overshoot percentage change (slight performance decline). The difference between the PID and the rest of the controllers seems to be that while other controllers achieved either improved performance results or small performance degradations when exposed to a decrease in inertial dynamics, the PID generally showed a large performance decline. The PID controller was specifically tuned for the initial controller evaluation on the waist joint, implying that it is over-optimised for its ideal conditions. Therefore, any dynamic changes to the system are expected to influence its performance negatively.

3) Gravity compensation vs. Additional payload: A similar trend as in Tab. VII is observed in comparing the gravity compensation test with the additional payload test in Tab. VIII.

TABLE VIII: Comparison of the gravity compensation test with an additional payload. **Blue bold** indicates the smallest performance percentage change, followed by **black bold**, while **orange bold** indicates the largest.

		PID	AIC	ReAIC	AFC
Δ ITAE (%)	mean	98.28	**17.62**	**81.92**	**102.61**
	std	93.59	**35.93**	**-13.16**	**175.88**
Δ t_s (%)	mean	**37.39**	**20.00**	20.10	**17.18**
	std	**142.17**	100.29	**47.10**	**-9.16**
Δ M_p (%)	mean	**-0.66**	445.04	**474.33**	**90.74**
	std	**-13.56**	**549.56**	331.79	**174.27**

Recognising that an increase in payload also constitutes an increase in the joint's mass moment of inertia, the same outcome was expected. Therefore, the same conclusions on increased payload can be made about each controller in the case of an increased mass moment of inertia.

The main anomalies are that the PID had the largest percentage change (largest performance decline) in settling time instead of the AIC, and the AIC had the second-largest percentage change (large performance decline) in overshoot. These differences can be attributed to the payload's gravitational influence instead of the mass moment of inertia change. The increase in settling time performance is due to the need for more control torque, therefore, the PID relies mostly on its integrator to cope with the payload's additional weight.

The increase in the AIC's overshoot was also observed in the qualitative analysis of the step response during the additional payload test in Section VI-F. Since the wrist was placed vertically during these payload tests, the AIC experienced more overshoot when the goal was the "zero-gravity position" (steps 3 and 7). This might suggest that the AIC is sensitive to abrupt gravitational changes.

The ReAIC's consistently superior performance over the AFC in gravity compensation and payload tests suggests robust adaptability in the presence of additional gravitational influences. However, as in the case of the AIC, a negative influence on the overshoot behaviour is expected.

VII. Discussion

A. Advantages and Limitations

This paper aimed to showcase how modifying the dynamic generative model of the AIC presented in [9] to that of Eq. (16), enhances the tunability, parameter influence and overall performance of the controller. The consequence of

this modification is the inclusion of one additional proportional gain tuning parameter $\mathcal{K}_p$, bringing the total number of tuning parameters to seven. While this may seem large, Section V-B shows that only three parameters require delicate tuning. The presence of $\mathcal{K}_p$ provides more control over performance metrics such as accuracy, settling time and overshoot, contrary to the AIC, for which this was previously impossible.

Section VI's experimental evaluation provided valuable insight into each controller's sensitivity, robustness and adaptability properties when exposed to different dynamic influences on the joints. The ReAIC consistently outperformed the AIC in all tests, with the exception of the overshoot performance metric. It also performed exceptionally well compared to the AFC, outperforming the AFC in the gravity compensation and payload tests. While the ReAIC did show signs of decreased performance in the presence of friction, it displayed robust and adaptive behaviour in all tests.

This paper provides substantial practical evidence supporting the stability and robustness of the ReAIC in handling non-linear dynamic changes. In Section IV-D, the ReAIC's state-space representation was introduced, facilitating the determination of the poles of the approximate closed-loop system, thus confirming its stability. Section V-B highlights the potential for instability resulting from significant increases in the learning rate κ_a of the control action. Such increases may cause overshooting during gradient descent operations, leading to control actions that surpass the minimum of the VFE. Thus, it is important to note that while a detailed formal proof of stability was not pursued within the scope of this paper, doing so would represent a valuable contribution and could be considered for potential future work.

In [12] and [13], strong emphasis is placed on the supposedly biased state estimation within the AIC framework. Yet, as highlighted in Sections II and IV, the reference state $\tilde{\boldsymbol{\mu}}$ does not represent the current state estimate, but rather the instantaneous temporal state at which the VFE is minimised. In other words, the manipulator's joint position y_q, velocity $y_{\dot{q}}$, goal state μ_g and the reference model dynamics are used to determine the most suitable desired reference state $\tilde{\boldsymbol{\mu}}$ that minimises the VFE. The control action ensures that the joint position and velocity are steered towards the reference state variables μ and μ', respectively.

The AIC and ReAIC were derived from an implicit mathematical simplification of the control action's gradient descent operation shown in Eq. (20). Here, the assumption is made that the relationship between the sensory input and the control action $\frac{\partial y}{\partial u}$ is linear and is represented by an identity matrix. This is an oversimplification, and a more accurate relationship representation may improve the overall performance of the controller. Past attempts [22] concluded that it increases the overall complexity of the framework and could produce unreliable results [9]. A suggestion might be that instead of modelling the true relationship, one could model observed relationships, for example, observing that below a certain action input u, the sensory position y_q does not change due to the inherent static friction within the joint.

B. Suggested Future Work

The experimental evaluations of the controllers unexpectedly highlighted the AFC's robustness and adaptability to cope with induced friction and dynamic inertial changes. The algorithm was modified to allow the AFC to cope with the influence of gravity (Appendix F). Future work could explore this new implementation and test its stability and adaptive properties.

A combination of the ReAIC and AFC could be attempted, as the AFC's remarkable friction compensation abilities would complement the ReAIC's performance sensitivity to friction. The controllers could be connected in parallel, or the AFC architecture could be embedded within the ReAIC's reference model to provide the desired reference behaviour. As observed in this paper, the selected reference model plays a vital role in the controller's success, therefore, more reference models should be explored and experimented with to discover this controller's potential further.

Finally, the AIC's limitations in fault-tolerant control were highlighted in [13], and as a result, the framework was adapted to better cope with faults, which produced the u-AIC [12]. A suggestion for future work would be to combine this framework, which emphasises the need to estimate the controller's current state, with the ReAIC's ability to determine the reference state that would best steer the controller towards the goal.

VIII. Conclusion

In this paper, we introduced the ReAIC, which is based on a previously implemented AIC. The ReAIC uses a neuroscience brain theory of the free-energy principle and active inference for online joint space control of a robotic manipulator. Our implementation modified the controller's dynamic reference model, provided clear tuning guidelines, and performed comparative experimental evaluations that displayed tunability, performance, and adaptability improvements over the AIC. The experimental evaluations showed that the ReAIC had superior performance in both gravity and payload compensation tests and the second-best overall performance in all other tests, for which the AFC outperformed all. Even though the ReAIC had large performance degradations in the presence of induced friction, it displayed stability, robustness and adaptability properties in all tests when compared to the other controllers. This work shows how a small modification can lead to large improvements, unveiling the hidden potential of active inference control. Since this is a relatively young controller, there are exciting prospects in which future work should explore alternative dynamic reference models or complementary combinations of the controllers.

Appendix A
Detailed Mathematical Derivations

Section III contained mathematical expressions in which it is not explicitly clear how the final result was achieved. In this appendix, we make step-by-step derivations of the most relevant equations with the help of Buckley et al. [14].

A. Derivation of the Kullback-Leibler expression: Eq. (3)

$$\begin{aligned} D_{KL}(r_d(x)||p(x|y)) &= \int r_d(x) \ln \frac{r_d(x)p(y)}{p(x,y)} \, \mathrm{d}x \\ &= \int r_d(x)(\ln \frac{r_d(x)}{p(x,y)} + \ln p(y)) \, \mathrm{d}x \\ &= \int r_d(x) \ln \frac{r_d(x)}{p(x,y)} \, \mathrm{d}x \\ &\quad + \int r_d(x) \ln p(y) \, \mathrm{d}x \\ &= \int r_d(x) \ln \frac{r_d(x)}{p(x,y)} \, \mathrm{d}x + \ln p(y) \\ &= \mathcal{F} + \ln p(y) \end{aligned} \tag{25}$$

Expanding $\mathcal{F}$ further result in:

$$\begin{aligned} \mathcal{F} &= \int r_d(x) \ln \frac{r_d(x)}{p(x,y)} \, \mathrm{d}x \\ &= \int r_d(x)(\ln r_d(x) - \ln p(x,y)) \, \mathrm{d}x \\ &= \int r_d(x) E(x,y) \, \mathrm{d}x + \int r_d(x) \ln r_d(x) \, \mathrm{d}x \end{aligned} \tag{26}$$

where

$$E(x,y) = -\ln p(x,y) \tag{27}$$

B. Derivation of $\mathcal{F}$ Eq. (5) from $r_d(x)$ Eq. (4)

Equation 4 formulates the R-density in terms of a Gaussian distribution. We will repeat this derivation here:

$$r_d(x) \equiv \mathcal{N}(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma}} \exp\{-\frac{(x-\mu)^2}{2\sigma}\}$$

For the upcoming derivation, notational clarity is desired by defining:

$$Z \equiv \sqrt{2\pi\sigma} \qquad \mathcal{E}(x) \equiv \frac{(x-\mu)^2}{2\sigma} \tag{28}$$

so that Eq. (4) can be simplified to:

$$r_d(x;\mu,\sigma) = \frac{1}{Z} \exp\{-\mathcal{E}(x)\} \tag{29}$$

With the newly defined R-density, further expanding Eq. (26) is necessary. Substituting Eq. (29) into the right integral of the variational free energy $\mathcal{F}$, the integral is solved by making use of the assumption that the R-density $r_d(x)$ normalises to [14]:

$$\int r_d(x) \, \mathrm{d}x = 1 \tag{30}$$

Recognising the expression for variance $\sigma = (x-\mu)^2$, the right integral of $\mathcal{F}$ in Eq. (26) can be expressed as [14]:

$$\begin{aligned} \mathcal{F} &= \ldots + \int r_d(x) \ln r_d(x) \, \mathrm{d}x \\ &= \ldots + \int r_d(x) \ln \frac{\exp\{-\mathcal{E}(x)\}}{Z} \, \mathrm{d}x \\ &= \ldots + \int r_d(x)(-\mathcal{E}(x) - \ln Z) \, \mathrm{d}x \\ &= \ldots - \ln Z - \int r_d(x)\mathcal{E}(x) \, \mathrm{d}x \\ &= \ldots - \frac{1}{2}(\ln 2\pi\sigma) - \int r_d(x) \frac{\cancelto{\sigma}{(x-\mu)^2}}{2\sigma} \, \mathrm{d}x \\ &= \ldots - \frac{1}{2}(\ln 2\pi\sigma) - \frac{1}{2} \end{aligned} \tag{31}$$

The left term of $\mathcal{F}$ demands further technical consideration due to the unspecified value of $E(x,y)$. Further assuming that the R-density $r_d(x)$ is sharply peaked at its mean value μ, such that the Gaussian bell shape is "squeezed" towards a delta function, and that $E(x,y)$ is a smooth function of x, the Taylor series expansion can be applied to $E(x,y)$ around its mean $x = \mu$ [14].

$$\begin{aligned} \mathcal{F} &= \int r_d(x) E(x,y) \, \mathrm{d}x + \ldots \\ &\approx \int r_d(x) \left\{ E(\mu,y) + \cancelto{0}{\left[\frac{\mathrm{d}E}{\mathrm{d}x}\right]_\mu} \delta x + \frac{1}{2}\left[\frac{\mathrm{d}^2E}{\mathrm{d}x^2}\right]_\mu \delta x^2 \right\} \mathrm{d}x + \ldots \\ &\approx E(\mu,y) + \frac{1}{2}\left[\frac{\mathrm{d}^2E}{\mathrm{d}x^2}\right]_\mu \int r_d(x) \delta x^2 \, \mathrm{d}x + \ldots \\ &\approx E(\mu,y) + \frac{1}{2}\left[\frac{\mathrm{d}^2E}{\mathrm{d}x^2}\right]_\mu \int r_d(x) \cancelto{\sigma}{(x-\mu)^2} \mathrm{d}x + \ldots \\ &\approx E(\mu,y) + \frac{1}{2}\left[\frac{\mathrm{d}^2E}{\mathrm{d}x^2}\right]_\mu \sigma + \ldots \end{aligned} \tag{32}$$

To arrive at the solution above, the normalisation assumption of the R-density is used, and from Eq. (27), it can be calculated that $\left[\frac{\mathrm{d}E}{\mathrm{d}x}\right]_\mu = 0$. Substituting the results of the two integrals Eqs. (31) and (32) back into Eq. (26), the variational free-energy $\mathcal{F}$ can be formulated as [14]:

$$\mathcal{F} = E(\mu,y) + \frac{1}{2}\left(\left[\frac{\mathrm{d}^2E}{\mathrm{d}x^2}\right]_\mu \sigma - (\ln 2\pi\sigma) - 1\right) \tag{33}$$

The variational free-energy $\mathcal{F}$ has now developed a dependency on the variance σ. This can be removed by demanding that $\frac{\mathrm{d}\mathcal{F}}{\mathrm{d}\sigma} = 0$ [14], as shown in Eq. (34), at which a constant term σ^* can be used.

$$\begin{aligned} \frac{\mathrm{d}\mathcal{F}}{\mathrm{d}\sigma} &= \frac{1}{2}\left\{\frac{\mathrm{d}}{\mathrm{d}\sigma}\left(\left[\frac{\mathrm{d}^2E}{\mathrm{d}x^2}\right]_\mu \sigma\right) - \frac{1}{\sigma}\right\} \\ &= \frac{1}{2}\left\{\left[\frac{\mathrm{d}^2E}{\mathrm{d}x^2}\right]_\mu - \frac{1}{\sigma}\right\} \end{aligned} \tag{34}$$

$$\sigma^* = \left[\frac{\mathrm{d}^2E}{\mathrm{d}x^2}\right]_\mu^{-1} \qquad \left[\frac{\mathrm{d}\mathcal{F}}{\mathrm{d}\sigma}\right]_{\sigma^*} \equiv 0 \tag{35}$$

Substituting σ^* into Eq. (33) and neglecting the constant term, an approximate for the variational free-energy $\mathcal{F}$ is obtained in terms of the approximate G-density $p(\mu, y)$, where the environmental states x have been replaced by the mean (best-guess) of the R-density μ [14].

$$\begin{aligned}\mathcal{F} &= E(\mu, y) - \frac{1}{2}\ln 2\pi\sigma^* \\ &\approx E(\mu, y) = -\ln p(\mu, y)\end{aligned} \tag{36}$$

This approximation suggests that an organism's internal representation of the environmental states μ only considers the most likely environmental causes, neglecting the details of their distributions.

Appendix B
Impulse Response Analysis

To determine an approximate transfer function of the manipulator waist joint, the dynamic model was assumed to be a second-order system:

$$m\ddot{x} + b\dot{x} = K_{\text{gain}}u \tag{37}$$

which equates to the following transfer function:

$$G(s) = \frac{K_{\text{gain}}}{ms^2 + bs} \equiv \frac{K_{\text{gain}}}{\tau s^2 + s} \tag{38}$$

Figure 15 shows the open loop impulse response of the waist joint after providing a PWM signal of 300 (4.07 V) for a duration of 0.4 s. Note that the increase in sampling frequency after 0.4 s is due to the control loop's termination at this time, thereby allocating additional data transfer bandwidth for sensory feedback. From the response, we were able

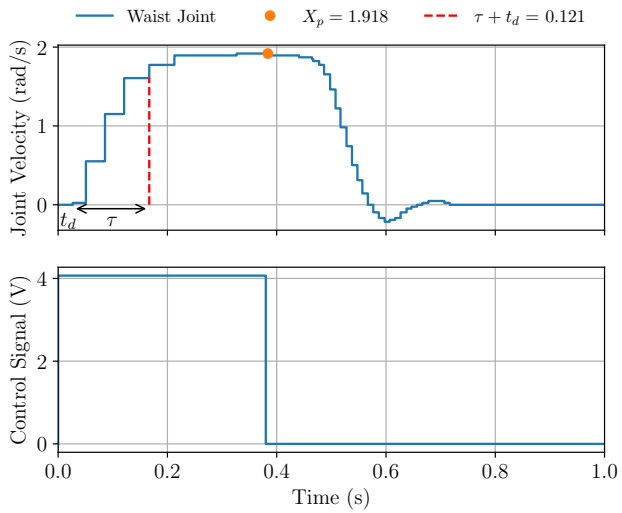


Fig. 15: Open-loop impulse response on the waist joint.

to determine the time constant τ (time at which joint velocity is $0.63X_p$) and the gain factor K_{gain}:

- $\tau = 0.0934$ s
- $K_{\text{gain}} = X_p/(u/u_{max}) = 5.659$

The approximate transfer function of the waist joint was calculated to be:

$$G(s) = \frac{5.659}{0.0934s^2 + s} \tag{39}$$

Appendix C
ReAIC State-Space Representation

The $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ matrices of the state variable form shown in Eqs. (21) and (22) are:

$$\mathbf{A} = [\mathbf{A}_1 \;\; \mathbf{A}_2 \;\; \mathbf{A}_3 \;\; \mathbf{A}_4]$$

$$\mathbf{A}_1 = \begin{bmatrix} -\kappa_\mu(\sigma_q^{-1} + \mathcal{K}_p\sigma_\mu^{-1}) \\ -\mathcal{K}_p\sigma_\mu^{-1}\kappa_\mu \\ 0 \\ \sigma_q^{-1}\kappa_a \end{bmatrix}$$

$$\mathbf{A}_2 = \begin{bmatrix} 1 - \sigma_\mu^{-1}\kappa_\mu \\ -\kappa_\mu(\sigma_{\dot{q}}^{-1} + \sigma_\mu^{-1} + \mathcal{K}_p\sigma_{\mu'}^{-1}) \\ -\mathcal{K}_p\sigma_{\mu'}^{-1}\kappa_\mu \\ \sigma_{\dot{q}}^{-1}\kappa_a \end{bmatrix}$$

$$\mathbf{A}_3 = \begin{bmatrix} 0 \\ 1 - \sigma_{\mu'}^{-1}\kappa_\mu \\ -\sigma_{\mu'}^{-1}\kappa_\mu \\ 0 \end{bmatrix}$$

$$\mathbf{A}_4 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \mathcal{K}_p\sigma_\mu^{-1}\kappa_\mu & \sigma_q^{-1}\kappa_\mu & 0 \\ \mathcal{K}_p\sigma_\mu^{-1}\kappa_\mu & 0 & \sigma_{\dot{q}}^{-1}\kappa_\mu \\ 0 & 0 & 0 \\ 0 & -\sigma_q^{-1}\kappa_\mu & -\sigma_{\dot{q}}^{-1}\kappa_a \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}$$

The corresponding ReAIC open-loop transfer function using the tuning parameters defined Section VI-A is:

$$\begin{aligned} C(s)_{\text{ReAIC}} &= \frac{1100s^2 + 2.97 \times 10^4 s + 1.513 \times 10^4}{s^4 + 177.5s^3 + 4190s^2 + 2175s} \\ &= \frac{1100(s + 26.48)(s + 0.5192)}{s(s + 149.6)(s + 27.38)(s + 0.5311)} \end{aligned} \tag{40}$$

which is a fourth-order transfer function. However, a pole and a zero are very close to each other, effectively reducing the order. This pole and zero are closely located due to the low confidence in the reference state variable $\sigma_{\mu'}$. Increasing $\sigma_{\mu'}$ further (decreasing the confidence) results in the pole and zero moving closer together, decreasing or eliminating its influence on the closed-loop system response.

Appendix D
PID Controller

This section provides a detailed overview of the PID controller implemented for this paper. The PID controller was defined as:

$$u = k_P e + k_I \int e dt + k_D \dot{e} \tag{41}$$

where

$$e = \mu_g - y_q$$
$$\dot{e} = 0 - y_{\dot{q}} \tag{42}$$

Here, the derivative control didn't use the change in the error but rather the velocity error, for which the desired velocity was set to zero. This was done to minimise noise due to the derivative of the error.

The controller transfer function is defined as:

$$C(s) = k_P + \frac{k_I}{s} + k_D s \tag{43}$$

A. Determining PID Tuning Parameters

The tuning parameters k_P, k_I and k_D were determined by utilising pole placement. The open-loop impulse response of the waist joint was used to measure the time constant τ and the gain factor K_{gain} in Appendix B.

Having defined the controller $C(s)$ and the plant $G(s)$ in Eq. (38), the closed-loop transfer function is defined as:

$$T(s)_{\text{cl}} = \frac{C(s)G(s)}{1 + C(s)G(s)} \tag{44}$$

for which the desired transfer function is:

$$T(s)_{\text{des}} = \frac{\omega_n^2}{(s^2 + 2\zeta\omega_n + \omega_n^2)(s + a)} \tag{45}$$

Substituting the values of C(s) and G(s) (Eqs. (43) and (38)) into the closed loop transfer function of Eq. (44), and by setting the characteristic equations of $T(s)_{\text{cl}} = T(s)_{\text{des}}$, we can equate the formulas for deducing the individual tuning parameters:

$$k_P = \frac{(\omega_n^2 + 2\zeta\omega_n a)\tau}{K_{\text{gain}}} \tag{46}$$

$$k_I = \frac{\omega_n^2 a \tau}{K_{\text{gain}}} \tag{47}$$

$$k_D = \frac{(2\zeta\omega_n + a)\tau - 1}{K_{\text{gain}}} \tag{48}$$

Pole placement was done by updating ω_n, ζ and a values until a satisfactory response was observed on the waist joint during step response tests. The final values are:

- $\omega_n = 4.09$
- $\zeta = 0.61$
- $a = 10.77$

The large value of a corresponds to the integral pole placement. This was required to allow the controller to overcome the inherent static friction within the joint much quicker. As a result, the final tuning parameters are:

- $k_P = 1.1705$
- $k_I = 2.9924$
- $k_D = 0.0851$

These were the tuning parameters used in all evaluation tests.

B. Integral Windup Prevention

A significant integral gain parameter k_I will almost always negatively influence the system's step response due to integral windup. In our initial experiments, it caused a large overshoot during each step response, especially the larger steps 1–3 and 7. Yet, it behaved as we intended when combatting the inherent static friction of the joint. To overcome this, a few prevention techniques were implemented in the control algorithm:

- The integral error e_I would only update once the error $e < 0.5$ rad. This resulted in a substantial decrease in overshoot while overcoming the static friction relatively quickly.
- If the control signal was already at its maximum PWM signal of 885 (12 V), the integral error e_I would stop updating.
- A cap was set on the maximum amount the integral error may sum up to.
- The integral error e_I would also stop updating if the error $e < 0.01$ rad and the joint velocity $y_{\dot{q}} = 0$ rad/s.

APPENDIX E
AIC CONTROLLER

The AIC was implemented using the suggested tuning procedure outlined in [9]. Since the tuning parameters do not influence the manipulator's rise time and overshoot response performance, it was tuned to display no oscillations. The following tuning parameters were used for all experiments:

- $\sigma_q = 1.0$
- $\sigma_{\dot{q}} = 1.0$
- $\sigma_\mu = 10.0$
- $\sigma_{\mu'} = 50.0$
- $\kappa_a = 5.0$
- $\kappa_\mu = 25.0$

The AIC framework is almost identical to that of the ReAIC, which is described in Section IV. The only exception is the proportional parameter $\mathcal{K}_p$ is always set to one.

APPENDIX F
AFC CONTROLLER

The AFC controller implemented within this paper was developed in [20]. Its control law uses a classic PD controller with an adaptive friction compensator w:

$$u = k_P e + k_D \dot{e} + w \tag{49}$$

where e and $\dot{e}$ are the same as in Eq. (42) and w is:

$$w = \hat{k}_c \sigma_1(\dot{x}, u) \tag{50}$$

Here, we need to define $\sigma_1(\dot{x}, u)$ and $\hat{k}_c$:

$$\sigma_1(\dot{x}, u) = \begin{cases} 1 & \text{if } \dot{x} > 0 \text{ or } (\dot{x} = 0 \text{ and } u > 0) \\ -1 & \text{if } \dot{x} < 0 \text{ or } (\dot{x} = 0 \text{ and } u < 0) \\ 0 & \text{if } \dot{x} = 0 \text{ and } u = 0 \end{cases} \tag{51}$$

$$\hat{k}_c = \hat{k}_{c(prev)} + \frac{\Delta t}{2}\dot{\hat{k}}_{c(prev)} + \frac{\Delta t}{2}\dot{\hat{k}}_c \tag{52}$$

and finally we can define $\dot{\hat{k}}_c$:

$$\dot{\hat{k}}_c = P\sigma_1(\dot{x}, u)(\dot{e} + \lambda e) \tag{53}$$

where P and λ are tuning parameters of the friction compensator.

A. Determining AFC Tuning Parameters

The PD tuning parameters k_P and k_D were determined using the same pole placement technique and open-loop impulse measurements as in Appendix D. In this case, since no integral gain is needed, this requires one less pole a. The formulas for the individual tuning parameters are:

$$k_P = \frac{\omega_n^2 \tau}{K_{\text{gain}}} \tag{54}$$

$$k_D = \frac{2\zeta\omega_n\tau - 1}{K_{\text{gain}}} \tag{55}$$

Pole placement was done by updating ω_n and ζ values until a satisfactory response was observed on the waist joint during step response tests. The final values are:

- $\omega_n = 7.92$
- $\zeta = 0.99$

From these parameters, the final PD tuning parameters were calculated to be:

- $k_P = 1.0413$
- $k_D = 0.0888$

λ is essentially the closed-loop pole location in the nominal situation; therefore, $\lambda = \omega_n\zeta$. P was increased until a satisfactory response was observed.

- $P = 0.5$
- $\lambda = 7.88$

.

B. AFC Modification for Gravity Compensation

The AFC implementation of the friction compensator requires the input $u = 0$ at its goal position. This doesn't work when the forces of gravity act on the system since the input may need to counteract the system's weight at its goal position. To overcome this and be able to use the AFC in both the gravity compensation and payload tests, a modified algorithm was implemented that uses a PID controller instead of a PD.

$$u = k_P e + k_I \int e\mathrm{d}t + k_D\dot{e} + w \tag{56}$$

and the $\sigma_1(\dot{x}, u)$ was altered to:

$$\sigma_1(\dot{x}, u) = \begin{cases} 1 & \text{if } \dot{x} > 0 \text{ or } (\dot{x} = 0 \text{ and } u > |k_I e_I|) \\ -1 & \text{if } \dot{x} < 0 \text{ or } (\dot{x} = 0 \text{ and } u < -|k_I e_I|) \\ 0 & \text{if } \dot{x} = 0 \text{ and } -|k_I e_I| \leq u \leq |k_I e_I| \end{cases} \tag{57}$$

where e_I is the accumulated integral error. This modification ensures that the friction compensation doesn't interfere with the control action generated by the integral control and only operates while $|k_P e + k_D\dot{e}| > 0$.

The final tuning parameters were selected to be the same as in the original AFC, with the additional k_I parameter set to $k_I = 1.5$.

The k_I is a smaller value than that of the PID controller in Appendix D, mainly because in the case of the PID controller, we relied on the integral action for a better steady-state error when exposed to the inherent static friction in the joint. In this case, we only wanted it to account for the gravitational influence, while the adaptive friction compensator would account for the frictional influence.

ACKNOWLEDGMENTS

I sincerely thank my supervisor Prof. Dr. Robert Babuska for his guidance and flexibility throughout this study. Additionally, I thank Dr. Corrado Pezzato, whose advice and explanations, drawn from his MSc Thesis on the AIC, helped shape this study.

REFERENCES

[1] K. J. Åström and B. Wittenmark, *Adaptive control*. Dover books on engineering, Mineola, N.Y: Dover Publications, 2nd ed., dover ed ed., 2008. OCLC: ocn196313137.

[2] S. G. Anavatti, F. Santoso, and M. A. Garratt, "Progress in adaptive control systems: past, present, and future," in *2015 International Conference on Advanced Mechatronics, Intelligent Manufacture, and Industrial Automation (ICAMIMIA)*, (Surabaya), pp. 1–8, IEEE, Oct. 2015.

[3] L. Pio-Lopez, A. Nizard, K. Friston, and G. Pezzulo, "Active inference and robot control: a case study," *Journal of The Royal Society Interface*, vol. 13, p. 20160616, Sept. 2016.

[4] K. Friston, J. Kilner, and L. Harrison, "A free energy principle for the brain," *Journal of Physiology-Paris*, vol. 100, pp. 70–87, July 2006.

[5] K. Friston, "The free-energy principle: a unified brain theory?," *Nature Reviews Neuroscience*, vol. 11, pp. 127–138, Feb. 2010.

[6] J. Kiverstein, M. D. Kirchhoff, and T. Froese, "The Problem of Meaning: The Free Energy Principle and Artificial Agency," *Frontiers in Neurorobotics*, vol. 16, p. 844773, June 2022.

[7] L. Da Costa, P. Lanillos, N. Sajid, K. Friston, and S. Khan, "How Active Inference Could Help Revolutionise Robotics," *Entropy*, vol. 24, p. 361, Mar. 2022.

[8] P. Lanillos, C. Meo, C. Pezzato, A. A. Meera, M. Baioumy, W. Ohata, A. Tschantz, B. Millidge, M. Wisse, C. L. Buckley, and J. Tani, "Active Inference in Robotics and Artificial Agents: Survey and Challenges," Dec. 2021. arXiv:2112.01871 [cs].

[9] 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, pp. 2973–2980, Apr. 2020.

[10] A. Cibiach Mercade, "Robot manipulator control under the Active Inference framework," *(Unpublished MSc thesis), TU Delft*, 2018.

[11] G. Oliver, P. Lanillos, and G. Cheng, "Active inference body perception and action for humanoid robots," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 14, pp. 462–471, June 2022. arXiv:1906.03022 [cs].

[12] M. Baioumy, C. Pezzato, R. Ferrari, and N. Hawes, "Unbiased Active Inference for Classical Control," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (Kyoto, Japan), pp. 12787–12794, IEEE, Oct. 2022.

[13] 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 *2021 European Control Conference (ECC)*, (Delft, Netherlands), pp. 1119–1125, IEEE, June 2021.

[14] 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, Dec. 2017.

[15] K. Friston, "Variational filtering," *NeuroImage*, vol. 41, pp. 747–766, July 2008.

[16] R. Zwanzig, *Nonequilibrium statistical mechanics*. Oxford ; New York: Oxford University Press, 2001.

[17] K. Friston, K. Stephan, B. Li, and J. Daunizeau, "Generalised Filtering," *Mathematical Problems in Engineering*, vol. 2010, pp. 1–34, 2010.

[18] C. Pezzato, "Active inference for adaptive and fault tolerant control: An application to robot manipulators," *(Unpublished MSc thesis), TU Delft*, 2019.

[19] G. F. Franklin, J. D. Powell, A. Emami-Naeini, and H. S. Sanjay, *Feedback control of dynamic systems*. Always learning, Boston, Mass.: Pearson, 7. ed., global ed ed., 2015.

[20] K. Verbert, R. Toth, and R. Babuska, "Adaptive Friction Compensation: A Globally Stable Approach," *IEEE/ASME Transactions on Mechatronics*, pp. 1–1, 2015.

[21] J. M. O'Kane, *A gentle introduction to ROS*. Columbia, SC: Jason M. O'Kane, version 2.1.1 (3e3d9c5), generated in november 20, 2014 ed., 2014.

[22] P. Lanillos and G. Cheng, "Active inference with function learning for robot body perception," *International Workshop on Continual Unsupervised Sensorimotor Learning*, 2018.