Adaptation through prediction
multisensory active inference torque control

Meo, Cristian; Franzese, Giovanni; Pezzato, Corrado; Spahn, Max; Lanillos, Pablo

DOI
10.1109/TCDS.2022.3156664

Publication date
2023

Document Version
Final published version

Published in
IEEE Transactions on Cognitive and Developmental Systems

Citation (APA)
Meo, C., Franzese, G., Pezzato, C., Spahn, M., & Lanillos, P. (2023). Adaptation through prediction: multisensory active inference torque control. *IEEE Transactions on Cognitive and Developmental Systems*, *15*(1), 32-41. https://doi.org/10.1109/TCDS.2022.3156664

Adaptation Through Prediction: Multisensory Active Inference Torque Control

Cristian Meo, Giovanni Franzese, Corrado Pezzato, Max Spahn, *Member, IEEE*, and Pablo Lanillos

***Abstract*—Adaptation to external and internal changes is of major importance for robotic systems in uncertain environments. Here, we present a novel multisensory active inference (AIF) torque controller for industrial arms that shows how prediction can be used to resolve adaptation. Our controller, inspired by the predictive brain hypothesis, improves the capabilities of current AIF approaches by incorporating learning and multimodal integration of low- and high-dimensional sensor inputs (e.g., raw images) while simplifying the architecture. We performed a systematic evaluation of our model on a 7DoF Franka Emika Panda robot arm by comparing its behavior with previous AIF baselines and classic controllers, analyzing both qualitatively and quantitatively adaptation capabilities and control accuracy. The results showed improved control accuracy in goal-directed reaching with high noise rejection due to multimodal filtering, and adaptability to dynamical inertial changes, elasticity constraints, and human disturbances without the need to relearn the model or parameter retuning.**

***Index Terms*—Bio-inspired robotics, robot control, representation learning.**

I. Introduction

Real-world complex robots, such as airplanes, cars, and manipulators, may need to process unstructured high-dimensional data coming from different sensors depending on the domain or task (e.g., LIDAR in cars, sonar in submarines, and different sensors to measure the internal state of the robotic system). In this context, one of the biggest challenges is mapping this rich stream of multisensory information into a lower dimensional space that integrates and compresses all modalities into a latent representation; the agent could then use this embedded latent representation that encodes the state of the robot and the world aiding the controller. Another key challenge is how to use this encoded representation to deal with real-world applications with changes and uncertainty. These environments may always present unmodeled behaviors, such as air turbulence in airplanes, unmodeled dynamics of water streams, or unexpected parameter changes. In the last years, some proof-of-concept studies in robotics have shown that active inference (AIF) may be a powerful framework to address challenges [18], such as adaptation [22], [28], robustness [1], [2], and multisensory integration [17], [20]. AIF is prominent in neuroscientific literature as a biologically plausible mathematical construct of the brain based on the free energy principle (FEP) [7]. According to this theory, the brain learns a generative model of the world/body that is used to perform state estimation (perception) as well as to execute control (actions), optimizing one single objective: Bayesian model evidence. This approach, which grounds on variational inference and dynamical systems estimation [9], has strong connections with Bayesian filtering [29] and control as inference [21], as it both estimates the system state and computes the control commands as a result of the inference process. Recent experiments in humans indicate that sensory prediction errors (SPE) may be responsible for body estimation and also involuntary adaptive active strategies that suppress multisensory conflicts [16]. Here, we show that once the robot has learned to predict the (multi)sensory input, then it can exploit those predictions to adapt to unexpected world/body variations, such as measurements noise, force disturbances, environmental changes (e.g., gravity or elasticity constraints), and internal changes (e.g., inertia or motor stiffness). We combine state representation learning [19] with variational free energy (VFE) optimization in generalized coordinates [8], [22] to infer the torques needed to achieve goal-directed behaviors. We evaluated our approach in several real-world experiments with a 7DoF Franka Emika Panda robot arm and compared it to state-of-the-art baselines in AIF and classic controllers.

A. Related Works

In 2003, Yamashita and Tani [30] described a robotic experiment that can be linked with the theory of what now is established as AIF [8]. They were able to generate motor primitives from sensorimotor experience in a top-down fashion. Since then, many researchers have pursued the design of these types of biologically (functional) plausible controllers [4]. Recently, a state estimation algorithm and an AIF-based reaching controller for humanoid robots were proposed in [15] and [22], respectively, showing robust sensory fusion (visual, proprioceptive, and tactile) and adaptability to unexpected sensory changes in real experiments. However, they could only handle low-dimensional inputs and did not implement low-level torque control. Latter, adaptive AIF torque controllers [2], [25] showed better performances than a state-of-the-art model

Manuscript received 16 September 2021; revised 13 December 2021; accepted 28 December 2021. Date of publication 3 November 2022; date of current version 13 March 2023. *(Corresponding author: Cristian Meo.)*

Cristian Meo, Giovanni Franzese, Corrado Pezzato, and Max Spahn are with the Faculty of Mechanical Engineering, Department of Cognitive Robotics, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: c.meo@student.tudelft.nl).

Pablo Lanillos is with Donders Institute for Brain, Cognition and Behaviour, Department of Artificial Intelligence, Radboud University, 6525 XZ Nijmegen, The Netherlands.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCDS.2022.3156664.

Digital Object Identifier 10.1109/TCDS.2022.3156664

reference adaptive controller. However, they cannot handle high-dimensional inputs. Furthermore, an AIF planning algorithm was presented in [12] and [27], showing that the introduction of visual working memory and the variational inference mechanism significantly improves the performance in planning adequate goal-directed actions. Reference [28] showed the plausibility of using neural networks architectures to scale AIF to raw images inputs. Finally, in the previous work, we presented a multimodal variational autoencoder active inference (MAIC-VAE) [20] torque controller, which integrated visual and joint sensory spaces. However, a clear and systematic comparison of adaptation between AIF and classic controllers is still missing. Besides, [20] did not present the generalized mathematical framework of multisensory AIF torque control scheme and the experiments were only in simulation.

B. Contribution

We describe the multisensory active inference controller (MAIC), which extends current AIF control approaches in the literature by allowing function learning [14], [17] through multimodal state representation learning [19] while maintaining the adaptation capabilities of an AIF controller and working at the level of torque. We provide the general mathematical framework of the MAIC and we derive two versions of the proposed algorithm as a proof of concept. Finally, we experimentally evaluated the proposed algorithm on a 7DOF Franka Emika Panda arm under different conditions. We systematically compared the MAIC with state-of-the-art torque AIF controllers, such as the AIC [25] and the uAIC [2], and standard controllers, such as model predictive control (MPC) and joint impedance control (IC). We present both qualitative and quantitative analysis in different experiments, focusing on adaptation capability and control accuracy.

II. AIF General Formulation and Notation

Here, we introduce the standard equations and concepts from the AIF literature [7], and the notation used in this article, framed for unimodal estimation and control of robotic systems [22]. The aim of the robot is to infer its state (unobserved variable) by means of noisy sensory inputs (observed). For that purpose, it can refine its state using the measurements or perform actions to fit the observed world to its internal model. This is dually computed by optimizing the VFE, a bound on the Bayesian model evidence [3]

System Variables: State, observations, actions, and their n-order time derivatives (generalized coordinates)

$$\begin{aligned}
\mathbf{x} &= [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_c], && \text{sensors observations } (c \text{ sensors})\\
\mathbf{r} &= [\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_c], && \text{sensory noise } (c \text{ sensors})\\
\tilde{\mathbf{x}} &= \left[\mathbf{x}, \mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n_d)}\right], && \text{generalized sensors}\\
\tilde{\mathbf{z}} &= \left[\mathbf{z}, \mathbf{z}^{(1)}, \dots, \mathbf{z}^{(n_d)}\right], && \text{multimodal system state}\\
\tilde{\boldsymbol{\mu}} &= \left[\boldsymbol{\mu}, \boldsymbol{\mu}^{(1)}, \dots, \boldsymbol{\mu}^{(n_d)}\right], && \text{proprioceptive state}\\
\tilde{\mathbf{r}} &= \left[\mathbf{r}, \mathbf{r}^{(1)}, \dots, \mathbf{r}^{(n_d)}\right], && \text{generalized sensory noise}\\
\tilde{\mathbf{w}} &= \left[\mathbf{w}, \mathbf{w}^{(1)}, \dots, \mathbf{w}^{(n_d)}\right], && \text{state fluctuations}\\
\mathbf{a} &= \left[a_1, a_2, \dots, a_p\right], && \text{actions } (p \text{ actuators}) \qquad (1)
\end{aligned}$$

where the notation $\mathbf{x}^{(n)} = (d^n\mathbf{x}/dt^n)$ is adopted for the nth-order derivative and n_d is the chosen number of generalized motions. Depending on the formulation, the action $\mathbf{a}$ can be force, torque, acceleration, or velocity. In this work, action refers to torque. We further define the time derivative of the state vector $D\tilde{\mathbf{z}}$ as

$$D\tilde{\mathbf{z}} = \frac{d}{dt}\left(\left[\mathbf{z}, \mathbf{z}', \dots, \mathbf{z}^n\right]\right) = \left[\mathbf{z}', \mathbf{z}'', \dots, \mathbf{z}^{n+1}\right].$$

Generative Models: Two generative models govern the robot: 1) the mapping function between the robot's state and the sensory input $g(\tilde{\mathbf{z}})$ (e.g., forward kinematics) and 2) the dynamics of the internal state $f(\tilde{\mathbf{z}})$ [3]

$$\tilde{\mathbf{x}} = g(\tilde{\mathbf{z}}) + \tilde{\mathbf{r}} \qquad (2)$$

$$D\tilde{\mathbf{z}} = f\left(\tilde{\mathbf{z}}\right) + \tilde{\mathbf{w}} \qquad (3)$$

where $\mathbf{r} \sim \mathcal{N}(\mathbf{0}, \Sigma_{\tilde{\mathbf{x}}})$ and $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \Sigma_{\tilde{\mathbf{z}}})$ are the sensory and process noise, respectively, and $\Sigma_{\tilde{\mathbf{x}}}$ and $\Sigma_{\tilde{\mathbf{z}}}$ are the covariance matrices that represent the controller's confidence about each sensory input and about its dynamics, respectively.

Variational Free Energy: The VFE is the optimization objective for both estimation and control. We use the definition of the $\mathcal{F}$ based on [8], where the action is implicit within the observation model $\mathbf{x}(a)$. Using the KL-divergence, the VFE is

$$\mathcal{F} = \mathrm{KL}\left[q\left(\tilde{\mathbf{z}}\right)||p\left(\tilde{\mathbf{z}}|\tilde{\mathbf{x}}\right)\right] - \log p(\tilde{\mathbf{x}}) \qquad (4)$$

where $q(\tilde{\mathbf{z}})$, $p(\tilde{\mathbf{z}}|\tilde{\mathbf{x}})$, and $p(\tilde{\mathbf{x}})$ are the variational density, posterior, and prior distribution. The VFE is an upper bound on the model evidence, and the minimization of the VFE will result in a minimization of surprise, and thus, a maximization of model evidence.

State estimation using gradient optimization

$$\dot{\tilde{\mathbf{z}}} = D\tilde{\mathbf{z}} - k_z \nabla_{\tilde{\mathbf{z}}}\mathcal{F}(\tilde{\mathbf{x}}, \tilde{\mathbf{z}}) \qquad (5)$$

Control using gradient optimization

$$\dot{\mathbf{a}} = -k_a \frac{\partial \tilde{\mathbf{x}}}{\partial a} \nabla_{\tilde{\mathbf{x}}}\mathcal{F}\left(\tilde{\mathbf{x}}, \tilde{\mathbf{z}}\right) \qquad (6)$$

where k_z and k_a are the gradient descent step sizes. The VFE has a closed form under the *Laplace and mean-field approximations* [3], [22] and it is defined as

$$\begin{aligned}
\mathcal{F}\left(\tilde{\mathbf{z}}, \tilde{\mathbf{x}}\right) &\triangleq -\ln p\left(\tilde{\mathbf{z}}, \tilde{\mathbf{x}}\right) - \frac{1}{2}\ln(2\pi|\Sigma|) \simeq -p\left(\tilde{\mathbf{x}}|\tilde{\mathbf{z}}\right)p\left(\tilde{\mathbf{z}}\right)\\
&\triangleq \left(\tilde{\mathbf{x}} - g\left(\tilde{\mathbf{z}}\right)\right)^T \Sigma_{\tilde{\mathbf{x}}}^{-1}\left(\tilde{\mathbf{x}} - g\left(\tilde{\mathbf{z}}\right)\right)\\
&\quad + \left(D\tilde{\mathbf{z}} - f\left(\tilde{\mathbf{z}}\right)\right)^T \Sigma_{\tilde{\mathbf{z}}}^{-1}\left(D\tilde{\mathbf{z}} - f\left(\tilde{\mathbf{z}}\right)\right)\\
&\quad + \frac{1}{2}\ln|\Sigma_{\tilde{\mathbf{x}}}| + \frac{1}{2}\ln|\Sigma_{\tilde{\mathbf{z}}}| \qquad (7)
\end{aligned}$$

where Σ is the optimal variance that optimizes the VFE [3]. The first two terms of (7) are the sensor and dynamics prediction error, while the last two are sensory and dynamics log variances (uncertainty associated).

Algorithm 1 MAIC

```
Require: $\mathbf{x}_d = \{\mathbf{x}_{d_1}, \mathbf{x}_{d_2}, \ldots, \mathbf{x}_{d_c}\}$
  while $\neg$goal reached do
      $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_c] \leftarrow c$ Sensors

      State Estimation
      $\dot{\tilde{\mathbf{z}}} \leftarrow$ multimodal state update law Eq. (9)

      Control Action
      $\dot{\mathbf{a}} = -\sum_{m=1}^{c} k_{\mathbf{a}_m} \partial_{\mathbf{a}} \mathbf{x}_m \Sigma_m^{-1} (\mathbf{x}_m - g_m(\tilde{\mathbf{z}}))$

      Euler Integration
      $\tilde{\mathbf{z}} \mathrel{+}= \delta_t \dot{\tilde{\mathbf{z}}}$
      $\mathbf{a} \mathrel{+}= \delta_t \dot{\mathbf{a}}$

  end while
```

Defining the Goal Through the Internal Dynamics: As in [28], we define the system internal dynamics $f(\tilde{\mathbf{z}})$ as

$$f\left(\tilde{\mathbf{z}}, \boldsymbol{\rho} = \mathbf{x}_d\right) = \frac{\partial \boldsymbol{g}\left(\tilde{\mathbf{z}}\right)}{\partial \tilde{\mathbf{z}}}\left(\mathbf{x}_d - \boldsymbol{g}\left(\tilde{\mathbf{z}}\right)\right) \tag{8}$$

where $\boldsymbol{\rho} = \mathbf{x}_d$ steers the system toward the desired target. In other words, the desired goal $\mathbf{x}_d$ produces an error with respect to the inferred state $\boldsymbol{g}(\tilde{\mathbf{z}})$, which causes an action toward $\mathbf{x}_d$ itself.

III. Architecture and Design: Multimodal Active Inference Controller

As long as we can learn the generative mapping of a certain sensory space, we can add any modality to (5), combining free energy optimization [8] with generative model learning and performing sensory integration. The online estimation and control problem is solved by optimizing the VFE through gradient optimization, computing (5) and (6). We first introduce the required preliminaries. Consequently, we illustrate the multimodal AIF update equations and the full algorithm.

A. Multimodal Active Inference

As discussed in [3], (7) can be extended for different modalities. Hence, state estimation and control equations can be derived for the multimodal case as well. We define the sensory generative function $g(\tilde{\mathbf{z}})$ with multiple modalities as $\boldsymbol{g}(\tilde{\mathbf{z}}) = [g_1(\tilde{\mathbf{z}}), \ldots, g_c(\tilde{\mathbf{z}})]$. Therefore, substituting (7) into (5) and (6) and rewriting it for the multimodal case, we can obtain the multimodal state estimation update law

$$\begin{aligned}\dot{\tilde{\mathbf{z}}} = D\tilde{\mathbf{z}} + \sum_{m=1}^{c}\left(k_m \frac{\partial g_m\left(\tilde{\mathbf{z}}\right)}{\partial \tilde{\mathbf{z}}} \Sigma_m^{-1}\left(\mathbf{x}_m - g_m\left(\tilde{\mathbf{z}}\right)\right)\right) \\ + k_{\mathbf{z}} \frac{\partial f\left(\tilde{\mathbf{z}}, \boldsymbol{\rho}\right)}{\partial \tilde{\mathbf{z}}} \Sigma_{\tilde{\mathbf{z}}}^{-1}\left(\mathbf{x}_d - f\left(\tilde{\mathbf{z}}, \boldsymbol{\rho}\right)\right)\end{aligned} \tag{9}$$

and the control equation

$$\dot{\mathbf{a}} = -\sum_{m=1}^{c} k_{\mathbf{a}_m} \partial_{\mathbf{a}} \mathbf{x}_m \Sigma_m^{-1}\left(\mathbf{x}_m - g_m\left(\tilde{\mathbf{z}}\right)\right) \tag{10}$$

where k_m and $k_{\mathbf{a}_m}$ are state estimation and control gradient descent step sizes related to modality m, and $\partial_{\mathbf{a}} \mathbf{x}_m = (\partial \mathbf{x}_m / \partial \mathbf{a})$. Algorithm 1 illustrates the general multimodal AIF controller scheme.

IV. Algorithm Implementations

In this work, we present two different implementations of the same algorithm as proofs of concept, chancing the dimensionality of the used sensory input. In the first case, we use end-effector positions (i.e., low-dimensional sensory inputs) $\mathbf{x}_{\mathbf{ee}}$, learning the generative mapping with Gaussian processes (MAIC-GP), while in the second case, we scale to the full raw images $\mathbf{x}_{\mathbf{v}}$ (i.e., high-dimensional sensory inputs), learning the mapping through a multimodal variational autoencoder (MAIC-VAE).

A. MAIC-GP

Here, we describe the multimodal AIF for low-dimensional inputs (e.g., end-effector position). We define the multisensory state and the sensory generative functions, respectively, as

$$\mathbf{x} = \left[\mathbf{x}_{\mathbf{q}}, \mathbf{x}_{\mathbf{ee}}\right] \tag{11}$$

$$g_{\boldsymbol{q}}(\boldsymbol{\mu}) = \boldsymbol{\mu} \tag{12}$$

$$g_{\mathbf{ee}}(\boldsymbol{\mu}) = GP_{\mathbf{ee}}(\boldsymbol{\mu}) \tag{13}$$

where $g_{\boldsymbol{q}}(\boldsymbol{\mu})$, as in [25], is the proprioceptive generative sensory function (i.e., joint states), and $g_{\mathbf{ee}}(\boldsymbol{\mu})$ is the end-effector generative sensory function. Since this implementation is a proof of concept and we are assuming that we do not know the system dynamics, as in [15], $g_{\mathbf{ee}}(\boldsymbol{\mu})$ is computed using a Gaussian Process (GP) regressor between proprioceptive sensory input and end-effector positions. This approach is particularly useful because we can compute a closed form for the derivative of the GP with respect to the beliefs $\boldsymbol{\mu}$, which is required for the multimodal state update law (9).

1) Learning: We train the model through guided self-supervised learning. This generated a data set of 9261 pairs end-effector positions and joint values $(\mathbf{X}_{\mathbf{ee}}, \mathbf{X}_{\mathbf{q}})$. We use a squared exponential kernel k of the form

$$k\left(\mathbf{x}_{\mathbf{q}_i}, \mathbf{x}_{\mathbf{q}_j}\right) = \sigma_f^2 e^{\left(-\frac{1}{2}\left(\mathbf{x}_{\mathbf{q}_i} - \mathbf{x}_{\mathbf{q}_j}\right)^T \boldsymbol{\Theta}\left(\mathbf{x}_{\mathbf{q}_i} - \mathbf{x}_{\mathbf{q}_j}\right)\right)} + \sigma_n^2 d_{ij} \tag{14}$$

where $\mathbf{x}_{\mathbf{q}_i}, \mathbf{x}_{\mathbf{q}_j} \in \mathbf{X}_{\mathbf{q}}$, d_{ij} is the Kronocker delta function, and $\boldsymbol{\Theta}$ is the hyperparameters diagonal matrix. We can compute the end-effector location given any joint state configuration as

$$g_{\mathbf{ee}}(\boldsymbol{\mu}) = k(\boldsymbol{\mu}, \mathbf{X}_q)\boldsymbol{K}^{-1}\mathbf{X}_{\mathbf{ee}}. \tag{15}$$

Finally, we can compute the derivative of $g_{\mathbf{ee}}(\boldsymbol{\mu})$ with respect to $\boldsymbol{\mu}$ as

$$\frac{\partial g_{\mathbf{ee}}(\boldsymbol{\mu})}{\partial \boldsymbol{\mu}} = -\boldsymbol{\Theta}^{-1}\left(\boldsymbol{\mu} - \mathbf{X}_q\right)^T\left[k\left(\boldsymbol{\mu}, \mathbf{X}_q\right)^T \cdot \boldsymbol{\alpha}\right] \tag{16}$$

where $\boldsymbol{K}$ is the covariance matrix, $\boldsymbol{\alpha} = \boldsymbol{K}^{-1}\mathbf{X}_{\mathbf{ee}}$ and $\cdot$ represents elementwise multiplication. Additional information about GP learning procedure can be found in Appendix B.

2) State Estimation and Control: Substituting (12) and (13) into (9) and (10), we can now write the state estimation update laws

$$\dot{\boldsymbol{\mu}} = \boldsymbol{\mu}^{(1)} + k_\mu \Sigma_{\boldsymbol{q}}^{-1} \boldsymbol{\epsilon}_{\mathbf{x}_{\mathbf{q}}} + k_{ee} \Sigma_{\mathbf{ee}}^{-1} \frac{\partial g_{\mathbf{ee}}(\boldsymbol{\mu})}{\partial \boldsymbol{\mu}} \boldsymbol{\epsilon}_{\mathbf{x}_{\mathbf{ee}}} - k_\mu \Sigma_{\boldsymbol{\mu}}^{-1} \boldsymbol{\epsilon}_{\boldsymbol{\mu}} \tag{17}$$

$$\dot{\boldsymbol{\mu}}^{(1)} = \boldsymbol{\mu}^{(2)} + k_\mu \Sigma_{\dot{\mathbf{q}}}^{-1}\boldsymbol{\epsilon}_{\dot{q}} - k_\mu \Sigma_{\boldsymbol{\mu}}^{-1}\boldsymbol{\epsilon}_{\boldsymbol{\mu}} - k_\mu \Sigma_{\boldsymbol{\mu}^{(1)}}^{-1}\boldsymbol{\epsilon}_{\boldsymbol{\mu}^{(1)}} \tag{18}$$

$$\dot{\boldsymbol{\mu}}^{(2)} = -k_\mu \Sigma_{\boldsymbol{\mu}^{(1)}}^{-1}\boldsymbol{\epsilon}_{\boldsymbol{\mu}^{(1)}} \tag{19}$$

where Σ_i^{-1} are the inverse variance (precision) matrices related to state observations and internal state beliefs and $\boldsymbol{\epsilon}_i$ are the SPE, with $i \in \{\mathbf{x_q}, \mathbf{x_{\dot{q}}}, \mathbf{x_{ee}}, \boldsymbol{\mu}, \boldsymbol{\mu}^{(1)}\}$. SPE represents the errors between expected sensory inputs and observed ones and are defined as

$$\begin{aligned}\boldsymbol{\epsilon}_{\mathbf{x_q}} &= \mathbf{x_q} - \boldsymbol{\mu} &(20)\\ \boldsymbol{\epsilon}_{\mathbf{x_{\dot{q}}}} &= \mathbf{x_{\dot{q}}} - \boldsymbol{\mu}^{(1)} &(21)\\ \boldsymbol{\epsilon}_{\mathbf{x_{ee}}} &= \mathbf{x_{ee}} - g_{\mathbf{ee}}(\boldsymbol{\mu}) &(22)\\ \boldsymbol{\epsilon}_{\boldsymbol{\mu}} &= \boldsymbol{\mu}^{(1)} + \boldsymbol{\mu} - \mathbf{x}_{\mathbf{q}_d} &(23)\\ \boldsymbol{\epsilon}_{\boldsymbol{\mu}^{(1)}} &= \boldsymbol{\mu}^{(1)} + \boldsymbol{\mu}^{(2)}. &(24)\end{aligned}$$

Finally, we can rewrite the control equation as

$$\dot{\mathbf{a}} = -k_a\bigg(\partial_{\mathbf{a}}\mathbf{x_q}\Sigma_q^{-1}\boldsymbol{\epsilon}_{\mathbf{x_q}} + \partial_{\mathbf{a}}\mathbf{x_{\dot{q}}}\Sigma_{\dot{\mathbf{q}}}^{-1}\boldsymbol{\epsilon}_{\mathbf{x_{\dot{q}}}} + \partial_{\mathbf{a}}\mathbf{x_{ee}}\frac{\partial g_{\mathbf{ee}}(\boldsymbol{\mu})}{\partial\boldsymbol{\mu}}\Sigma_{\mathbf{ee}}^{-1}\boldsymbol{\epsilon}_{\mathbf{x_{ee}}}\bigg). \tag{25}$$

Note that as in [25], in (25), the partial derivatives with respect to the action are set to identity matrices, encoding just the sign of the relation between actions and the change in the observations. Although we can compute the action inverse models $\partial_{\mathbf{a}}\mathbf{x_q}$, $\partial_{\mathbf{a}}\mathbf{x_{\dot{q}}}$, $\partial_{\mathbf{a}}\mathbf{x_{ee}}$ through online learning using regressors [14], we let the adaptive controller absorb the nonlinearities. Thus, as described by [25], we just consider the sign of the derivatives.

B. MAIC-VAE

Here, we describe the multimodal AIF controller for high-dimensional sensory inputs. We use the autoencoder architecture to compress the information into a common latent space $\mathbf{z}$ that represents the system's internal state. We define the multisensory state and sensory generative functions, respectively, as

$$\begin{aligned}\mathbf{x} &= [\mathbf{x_q}, \mathbf{x_v}] &(26)\\ g_q(\mathbf{z}) &= \text{decoder}_{\mathbf{q}}(\mathbf{z}) &(27)\\ g_{\mathbf{v}}(\mathbf{z}) &= \text{decoder}_{\mathbf{v}}(\mathbf{z}) &(28)\end{aligned}$$

where $\text{decoder}_{\mathbf{q}}(\mathbf{z})$ and $\text{decoder}_{\mathbf{v}}(\mathbf{z})$ describe the mapping between $\mathbf{z}$ and the sensory spaces. The interested reader can find a detailed description of MAIC-VAE in [20].

1) Generative Models Learning: The multimodal variational autoencoder (MVAE) was trained through guided self-supervised learning. The data set generated (50 000 samples) consisted in pairs of images with size (128 × 128) and joint angles $(\mathbf{X_v}, \mathbf{X_q})$. In order to accelerate the training, we included a precision mask $\Pi_{\mathbf{x_v}} = \Sigma_{\mathbf{x_v}}^{-1}$, computed by the variance of all images and highlighting the pixels with more information. The augmented reconstruction loss employed was

$$\mathcal{L} = \text{MSE}\big((1 + \Pi_{\mathbf{x_v}})g_{\mathbf{v}}(\mathbf{z}),\ \mathbf{x_v}\big) + \text{MSE}\big(g_{\mathbf{q}}(\mathbf{z}), \mathbf{x_q}\big) \tag{29}$$

where $\mathbf{x_q} \in \mathbf{X_q}$ and $\mathbf{x_v} \in \mathbf{X_v}$. Appendix C provides a detailed description of the MVAE learning procedure.

2) State Estimation and Control: As in MAIC-GP, substituting the defined generative mappings, (27) and (28), into (9) and (10), we can rewrite the *state estimation* update law

$$\dot{\mathbf{z}} = k_v\frac{\partial g_{\mathbf{v}}}{\partial\mathbf{z}}\Sigma_{\mathbf{x_v}}^{-1}(\mathbf{x_v} - g_{\mathbf{v}}(\mathbf{z})) + k_q\frac{\partial g_{\mathbf{q}}}{\partial\mathbf{z}}\Sigma_{\mathbf{q}}^{-1}(\mathbf{x_q} - g_{\mathbf{q}}(\mathbf{z})) - k_z\frac{\partial f}{\partial\mathbf{z}}\Sigma_f^{-1}(\mathbf{x}_d - f(\mathbf{z}, \boldsymbol{\rho})). \tag{30}$$

As we do not have access to the high-order generalized coordinates of the latent space $\mathbf{z}'$ and $\mathbf{z}''$, we track both the multimodal shared latent space $\mathbf{z}$ and the higher orders of the proprioceptive (joints) state $\boldsymbol{\mu}^{(1)}$ and $\boldsymbol{\mu}^{(2)}$. Thus, we update the proprioceptive state velocity and acceleration using (18) and (19), while the joint angles are predicted by the MVAE: $\boldsymbol{\mu} = g_{\mathbf{q}}(\mathbf{z})$. Finally, as before the *action* (torque) is computed by optimizing the VFE using (6). Here, since we cannot easily compute the partial derivative of $g_{\mathbf{v}}$ with respect to the action, we only consider the proprioceptive errors. Thus, the torque commands are updated with the following differential equation:

$$\dot{\mathbf{a}} = -k_a\Big(\Sigma_{\mathbf{q}}^{-1}\boldsymbol{\epsilon}_{\mathbf{x_q}} + \Sigma_{\dot{\mathbf{q}}}^{-1}\boldsymbol{\epsilon}_{\mathbf{x_{\dot{q}}}}\Big) \tag{31}$$

where even in this case we just consider the sign of the partial derivatives $\partial_{\mathbf{a}}\boldsymbol{\mu}$, $\partial_{\mathbf{a}}\boldsymbol{\mu}^{(1)}$.

V. Results

A. Experiments and Evaluation Measures

We systematically evaluated our MAIC approach in a 7DOF Franka Emika Panda robot arm. We performed three different experimental analyzes and compared the MAIC approach against two state-of-the-art torque AIF controllers (AIC [25] and uAIC [2]) and two classic controllers: 1) (MPC, Appendix A) and 2) (IC, Appendix E).

1) *Qualitative Analysis in Sequential Reaching (Section V-C):* We evaluated MAIC approaches qualitative behaviors, focusing on how multimodal filtering affects control accuracy on the presented controllers.
2) *Adaptation Study (Section V-D):* We evaluated the response of the system to unmodeled dynamics and environment variations by altering dynamically the mass matrix (inertial experiment), by adding an elastic constraint (constrain experiment), by adding random human disturbances (human disturbances experiment), and by adding random noise to the published joints values (noisy experiment).
3) *Ablation Analysis in Sequential Reaching (Section V-C):* We evaluated the algorithm's accuracy and behavior removing the extra modality from the algorithm.

In order to evaluate the experiments, we used the following evaluation metrics.

1) *Joints Perception Error:* It is the error between the inferred (belief) and the observed joint angle. The more accurate the predictions are, the lower will be the perception error.
2) *Joints Goal Error:* It is the error between the current joint angles and the desired ones (goal).

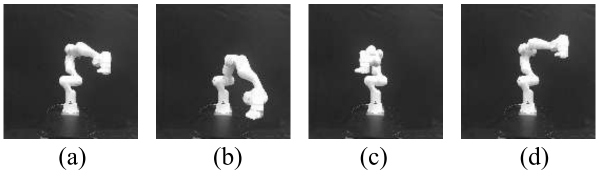


Fig. 1. Goal poses images. (a) $\mathbf{x}_{\mathbf{v}_{d1}}$. (b) $\mathbf{x}_{\mathbf{v}_{d2}}$. (c) $\mathbf{x}_{\mathbf{v}_{d3}}$. (d) $\mathbf{x}_{\mathbf{v}_{d4}}$.

3) *Image Reconstruction Error:* It is the error between the predicted visual input and the observed image. It is computed as the Frobenius norm of the difference between current and goal images. It describes the accuracy of the visual generative model.

4) *End-Effector Reconstruction Error:* It is the Euclidean distance between the predicted end-effector positions and the ones computed through the forward kinematics of the observed joints.

To summarize, joints perception and image reconstruction errors measure how well the state is estimated, while joints goal errors give a measure of how well the control task is executed.

B. Experimental Setup and Parameters

The experiments were performed on the 7DOF Franka Panda robot arm using ROS [13] as the interface, Pytorch [23] for the MVAE, and Sklearn [24] for the GPs. An Intel Realsense D455 camera was used to acquire visual gray scaled images with size of 128×128 pixels. The camera was centered in front of the robot arm with a distance of 0.8 m.

The tuning parameters for the MAIC controllers are as follows.

1) $\Sigma_{\mathbf{x}_\mathbf{v}}$*:* Variance representing the visual sensory data confidence, which was set as the variances of the training data set.

2) $\delta_t = 0.001$*:* Euler integration step.

3) $\Sigma_{\mathbf{q}} = 3, \Sigma_{\dot{\mathbf{q}}} = 3, \Sigma_{\boldsymbol{\mu}} = 5, \Sigma_{\boldsymbol{\mu}^{(1)}} = 5, \Sigma_f = 4,$ and $\Sigma_{\mathbf{ee}} = 6$*:* Variances representing the confidence of internal belief about the states.

4) $k_\mu = 18.67, k_q = 1.5, k_v = 0.2, k_{ee} = 1.4,$ and $k_a = 9$*:* The learning rates for state update and control actions respectively, were manually tuned in the ideal settings experiment.

All experiments were executed on a computer with CPU: Intel Core i7 8th Gen, GPU: Nvidia GeForce GTX 1050 Ti.[1]

C. Qualitative Analysis in Sequential Reaching Task

In order to analyze MAIC qualitative behavior, we designed a sequential reaching task with desired goals $\mathbf{x}_d = [\mathbf{x}_{\mathbf{q}_d}, \mathbf{x}_{\mathbf{ee}_d}]$ and $\mathbf{x}_d = [\mathbf{x}_{\mathbf{q}_d}, \mathbf{x}_{\mathbf{v}_d}]$, respectively, defined for MAIC-GP and MAIC-VAE. The sequential reaching task is evaluated using four different desired states, defined by the final joint angles $\{\mathbf{x}_{\mathbf{q}_{d1}}, \mathbf{x}_{\mathbf{q}_{d2}}, \mathbf{x}_{\mathbf{q}_{d3}}, \mathbf{x}_{\mathbf{q}_{d4}}\}$, expressed in radiants as follows.

1) $\mathbf{x}_{\mathbf{q}_{d1}} = [0.45, -0.38, 0.32, -2.45, 0.14, 2.06, 1.26]$.

2) $\mathbf{x}_{\mathbf{q}_{d2}} = [0.70, -0.15, 0.10, -2.65, 0.31, 2.55, 1.23]$.

3) $\mathbf{x}_{\mathbf{q}_{d3}} = [-0.03, -0.73, -0.25, -2.69, -0.18, 1.83, 0.79]$.

4) $\mathbf{x}_{\mathbf{q}_{d4}} = [0.31, -0.47, 0.38, -2.16, 0.14, 1.71, 1.28]$.

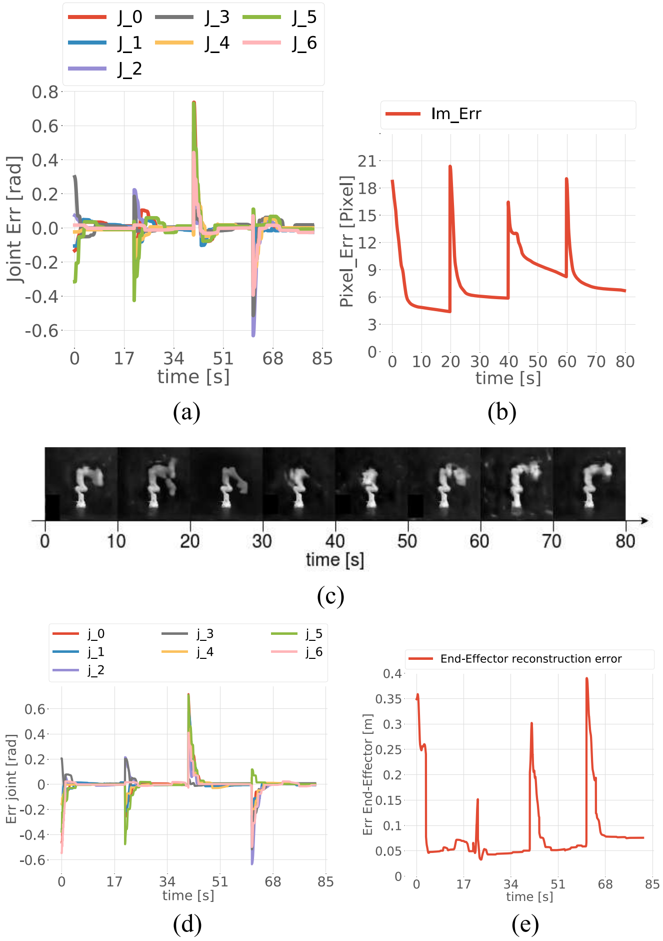


Fig. 2. Qualitative analysis of the error measures in the sequential reaching of four goals. All errors present peaks when a new goal is set. (a) and (d) Each line represents the error between the ith joint belief and the ground truth. (b) Image reconstruction error. (c) Sequence of the predicted images by the generative model along the trajectory. (e) End-effector Reconstruction error.

The desired end-effector positions $\{\mathbf{x}_{\mathbf{ee}_{d1}}, \mathbf{x}_{\mathbf{ee}_{d2}}, \mathbf{x}_{\mathbf{ee}_{d3}}, \mathbf{x}_{\mathbf{ee}_{d4}}\}$ and the desired visual input $\{\mathbf{x}_{\mathbf{v}_{d1}}, \mathbf{x}_{\mathbf{v}_{d2}}, \mathbf{x}_{\mathbf{v}_{d3}}, \mathbf{x}_{\mathbf{v}_{d4}}\}$, where both desired end-effector positions and visual input are defined consistently with the desired joint positions (Fig. 1). In order to select unbiased desired goals, all the desired joint poses were randomly sampled from the data set. In all experiments, the robot starts from the home position ($\mathbf{x}_{\mathbf{q}_{\text{home}}} = \mathbf{x}_{\mathbf{q}_{d4}}$rad).

1) MAIC-VAE Qualitative Behavior: Fig. 2(a)–(c) illustrates MAIC-VAE qualitative internal behavior. It can be seen that both modalities are successfully estimated. However, Fig. 2(a) shows that joints reconstructions present overshoot, leading to a similar behavior on the control task, as shown in Fig. 3. Moreover, the robot updates its internal belief by approximating the conditional density, maximizing the likelihood of the observed sensations, and then generates an action that results in a new sensory state, which is consistent with the current internal representation. However, the visual decoder requires much more computational time than the main control loop, leading to the irregular behavior shown in Fig. 2(a). Although Fig. 2(b) shows that image reconstructions present different errors for different poses, Fig. 2(c) shows that the image reconstructions through the experiment are well reconstructed.

[1] For reproducibility, the code is publicly available at https://github.com/Cmeo97/MAIC.

TABLE I

QUANTITATIVE JOINTS GOAL ERRORS COMPARISON. RMSE [RAD] AND STD [RAD] OF THE JOINTS ERRORS ARE PRESENTED, LOWEST ERRORS ARE SHOWED IN *Black Bold* AND SECOND LOWEST IN *Blue Bold*. ERRORS ARE COMPUTED FOR THE FULL EXPERIMENT, TRANSIENT PHASE (0–10 S) AND STEADY STATE (10–20 S)

	Controllers	Vanilla Experiment		Inertial Experiment		Constraint Experiment		Human disturbances Exp		Noisy Experiment	
		RMSE	std	RMSE	std	RMSE	std	RMSE	std	RMSE	std
Full Experiment	AIC	4.04E-03	4.85E-03	7.23E-03	3.05E-02	5.41E-03	1.42E-02	4.07E-03	1.21E-02	4.91E-03	3.33E-02
	uAIC	3.28E-03	1.32E-02	**3.38E-03**	1.16E-02	4.10E-03	8.88E-03	3.32E-03	9.56E-03	**3.03E-03**	2.20E-02
	MAIC-VAE	**3.18E-03**	1.78E-02	3.40E-03	1.45E-02	**3.65E-03**	2.26E-02	**3.62E-03**	1.44E-02	**2.38E-03**	1.81E-02
	MAIC-GP	**3.09E-03**	1.71E-02	**3.33E-03**	1.89E-02	**3.20E-03**	1.50E-02	**3.13E-03**	2.20E-02	3.40E-03	1.91E-02
	MPC	2.41E-02	6.81E-03	4.43E-02	1.77E-02	3.31E-02	7.84E-03	2.20E-01	5.00E-02	4.95E-02	1.32E-02
	IC	9.45E-03	2.07E-02	1.95E-02	1.87E-02	1.54E-02	1.23E-02	9.76E-03	2.04E-02	4.84E-03	2.13E-02
Transient	AIC	8.09E-03	3.97E-02	9.67E-03	4.18E-02	9.94E-03	1.97E-02	8.14E-03	1.68E-02	9.76E-03	4.22E-02
	uAIC	6.54E-03	1.85E-02	**6.75E-03**	1.62E-02	8.03E-03	1.24E-02	6.62E-03	1.33E-02	8.98E-03	2.50E-02
	MAIC-VAE	**6.36E-03**	2.48E-02	6.76E-03	2.02E-02	**7.26E-03**	3.15E-02	**6.48E-03**	2.01E-02	**6.63E-03**	2.71E-02
	MAIC-GP	**6.18E-03**	2.38E-02	**6.63E-03**	2.63E-02	**6.40E-03**	2.09E-02	**6.26E-03**	3.03E-02	**6.78E-03**	2.66E-02
	MPC	3.12E-02	9.45E-03	7.04E-02	2.47E-02	5.17E-02	1.09E-02	2.23E-01	4.96E-02	3.36E-02	1.82E-02
	IC	1.63E-02	2.89E-02	3.48E-02	2.62E-02	2.72E-02	1.72E-02	1.69E-02	2.86E-02	1.86E-02	2.98E-02
Steady-state	AIC	**1.77E-06**	1.84E-06	4.88E-05	6.30E-07	8.70E-04	1.50E-03	**1.77E-06**	8.79E-05	8.33E-05	7.37E-04
	uAIC	**1.19E-05**	1.14E-05	**1.26E-05**	1.86E-05	1.69E-04	2.79E-04	3.201E-05	3.32E-05	5.89E-04	7.38E-03
	MAIC-VAE	3.29E-05	2.97E-05	3.50E-05	4.25E-05	**3.55E-05**	4.16E-05	3.31E-05	3.71E-05	**4.04E-05**	3.35E-04
	MAIC-GP	1.66E-05	2.02E-05	**1.77E-05**	2.47E-05	**1.54E-05**	8.67E-05	**1.69E-05**	3.21E-03	**7.15E-05**	4.90E-04
	MPC	1.70E-02	1.54E-03	1.81E-02	1.75E-03	1.44E-02	1.26E-03	1.18E-01	5.04E-02	1.81E-02	3.19E-03
	IC	2.61E-03	2.55E-03	4.32E-03	3.57E-04	3.64E-03	2.45E-03	2.62E-03	2.70E-03	2.91E-03	5.07E-03

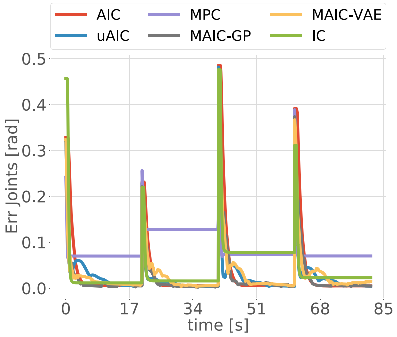


Fig. 3. Vanilla comparison. Lines represent the average of absolute joints goal errors. Peaks are present when the new goal is set.

2) MAIC-GP Qualitative Behavior: Fig. 2(d) and (e) illustrates MAIC-GP qualitative internal behavior. As in the previous case, both modalities are successfully estimated. Fig. 2(d) shows that MAIC-GP joint estimations do not overshoot.

3) Vanilla Comparison: Fig. 2 illustrates the qualitative behavior of the compared controllers. From one goal to the next one, the errors drop down. Although the joint belief errors [Fig. 2(a)] show synchronous convergence without significant steady-state errors, due to slow algorithmic frequency, the MVAE-AIC behavior is not smooth.

Moreover, some goals can be better reconstructed than others, resulting in different steady-state errors. The reason is that different **z** solutions lead to similar images. Furthermore, due to dynamical model errors, MPC and IC present significant steady-state errors. Finally, MAIC-VAE and uAIC overshoot, while all the other present overdamped behaviors.

D. Adaptation Study

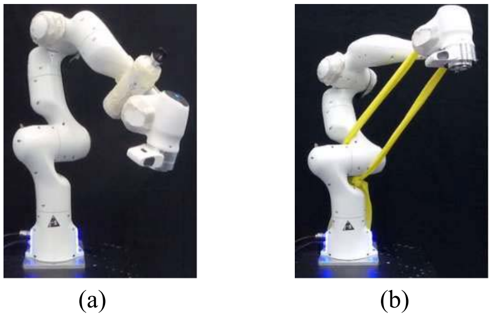
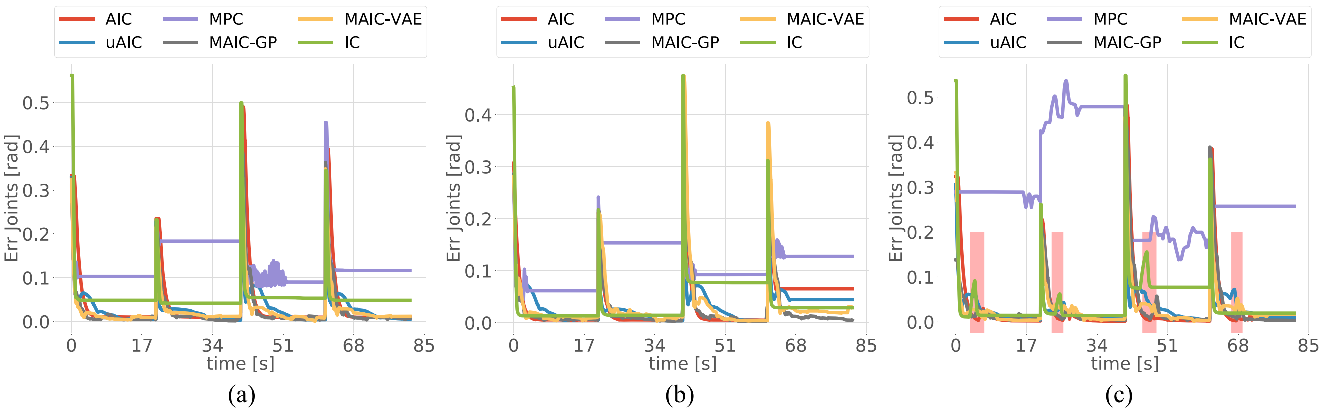


Fig. 4. Experimental setup. (a) Inertial experiment: a bottle half full of water is attached to the 5th joint. (b) Constraint Experiment setup: an elastic band links the first to the 5th joint.

To investigate our approach adaptability to unmodeled dynamics and environment variations we systematically tested the controllers in four experiments. The first three experiments aim to evaluate the adaptability to unmodeled dynamics and the robustness against variations on inertial parameters. First, we attached a bottle half full of water to the 5th joint [Fig. 4(a)]. As a result, due to water movements, the robot inertia changes dynamically. Second, we constrained the robot with an elastic band [Fig. 4(b)], connecting the first robot link to the last one and, therefore, introducing a substantial change in the robot dynamics. Third, we perturbed the robot along the experiment pushing it along random directions and, therefore, testing if they are able to recover from human random disturbances. Finally, we reevaluated the controllers in the presence of sensory noise, focusing on the robot behavior. Again, we compared our algorithm implementations (MAIC-GP and MAIC-VAE) with AIC, uAIC, MPC, and an IC. All controllers parameters were the same as in the previous experiments: no retuning was done. Table I reports the root-mean-square errors (RMSE) and the related standard deviations (std), which represent all the results collected during the experiments, the most accurate results are highlighted in black bold and the second most accurate in blue bold. In



(a) (b) (c)

Fig. 5. Lines represent the average of absolute joints goal errors. Peaks coincide with the instants when a new goal is set, overshoots instead are present some seconds later, when the error already dropped substantially. (a) Inertial Experiment. (b) Constraint Experiment. (c) Red rectangles show the time intervals on which the disturbances are applied, small peaks represent human disturbances.

order to evaluate quantitatively both steady-state errors, transient behavior, and average errors, we present both RMSE and std for each phase. On average, MAIC-GP is the most robust against dynamic parameters change and the most adaptive to unmodeled dynamics, while MAIC-VAE is the best one on noise rejection. Only at the steady-state (after 10 s of execution) AIC has the lowest error on both Vanilla and Human disturbances experiments and uAIC at inertial experiment due to its integration term. Furthermore, at the steady state, MAIC-GP adapts better in the constraint experiment and MAIC-VAE is the best one on noise rejection. Finally, although both MPC and IC reported the worst performances in all experiments, they presented significant offsets already in the vanilla comparison. Therefore, we will focus just on their qualitative behaviors. We now present the details of each experiment.

1) Inertial Experiment: A bottle half full of water has been attached to the 5th robot joint. The water moves along the experiment, changing the inertial characteristic of the object attached to the robot. Fig. 5(a) illustrates the controllers' qualitative behaviors during the inertial experiment. It can be seen that due to the unmodeled dynamics, IC and MPC show different offsets than the ones in the vanilla comparison. Moreover, MPC shows an unstable behavior in one of the desired poses. Furthermore, since all the AIF controllers do not use any robot model, they are not affected by the change of dynamics. Table I shows that on average the most accurate controllers are MAIC-GP (3.33E-03), uAIC (3.38E-03), and MAIC-VAE (3.40E-03).

2) Elastic Constraint Experiment: The experiment aims to drastically change the underlying dynamics of the system. Specifically, a rubber band was attached to the robot. To prevent the robot from entering safety mode, we chose to link the first joint to the last one. We bounded the elastic tension to a sustainable value. Fig. 5(b) shows that both classic and unimodal AIF controllers are significantly affected by the elastic tension, presenting remarkable offsets. In contrast, as recorded on Table I, MAIC-GP and MAIC-VAE present the highest control accuracy.

3) Human Disturbances Experiment: This experiment aims to evaluate compliance and controllers' recovery ability after random disturbances. To do this, a human operator pushed the robot in random directions during the experiment. Red shaded areas in Fig. 5(c) indicate the periods on which the robot is disturbed. Apart from the MPC, which is not able to recover and perform the task, all the other ones fully recover from the disturbances, showing a safe behavior in case of human disturbances.

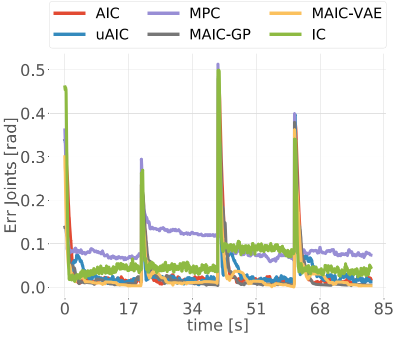


Fig. 6. Noisy experiment. Lines represent the average of absolute joints goal errors. Peaks coincide with the instants when a new goal is set.

4) Noise Experiment: We reevaluated the controller behavior in the presence of proprioceptive noise, focusing on the noise rejection capabilities of the six controllers. Proprioceptive noise was implemented as additive noise sampled from a normal distribution $\mathbf{r_q} \sim \mathcal{N}(\mathbf{0}, \Sigma_{\mathbf{r_q}} = 0.1)$. Fig. 6 shows that MAIC controllers were the most adaptive, presenting the smoothest behaviors. The reason is that multimodal filtering acts as a filter for the injected noise, reducing its effect and allowing a smooth control behavior. All the other controllers oscillate significantly more along the experiment.

E. Ablation Study

In order to evaluate the effect of the extra modalities, we performed an ablation study removing the extra modality from the algorithm scheme. Fig. 7 shows that by removing the visual modality, the behavior becomes much smoother. Indeed, the control loop frequency increased from 120 to 1000 Hz. However, Table II reports that the control accuracy does not change significantly. Moreover, from Fig. 7, it can be seen that

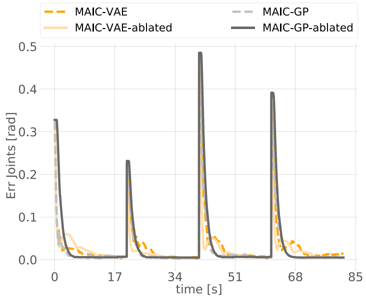


Fig. 7. Ablation study. Lines represent the average of absolute joints goal errors. Peaks are present when the new goal is set.

TABLE II

Ablation Study: Quantitative Analysis. RMSE [Rad] and Std[Rad] Are Shown for Both MAICs and Their Ablated Versions. Lowest Errors Are Showed in *Black Bold* and Second Lowest in *Blue Bold*

		RMSE [rad]	std [rad]
Full Experiment	MAIC-VAE	**3.18E-03**	1.78E-02
	MAIC-VAE-ablated	3.21E-03	1.36E-02
	MAIC-GP	**3.09E-03**	1.71E-02
	MAIC-GP-ablated	4.04E-03	4.84E-03

controllers' response behaviors do not change when they are ablated.

VI. Limitations and Advantages

On the one hand, although the quantitative table comparison shows that on average MAIC implementations are more adaptive and accurate, they still have limitations. First, multimodal filtering requires more computational time, leading to irregular behaviors. Indeed, the ablation study clearly shows that when removing the visual modality, the control behavior becomes significantly smoother. Using a faster GPU may solve this issue. Moreover, the multimodal state estimation depends on the accuracy of the learned generative mapping. In all experiments, we used a black background to facilitate the image reconstruction. Furthermore, another limitation is that for goal-directed behaviors, we need to provide the desired values for all the sensor modalities, which may not be always available. However, as in [26], it may be possible to combine MAIC with a high-level controller in order to control complex robotics systems (e.g., soft robots). On the other hand, MAIC can incorporate any type and number of sensors besides the end-effector position or images. It can work in an imaginary regime (Appendix D) by mentally simulating the expected behavior, opening many opportunities for future research such as model predictive AIF controllers, where the controller predicts N steps head. Besides, the multimodal filtering scheme can be integrated into other kinds of controllers, such as an IC.

VII. Conclusion

We described MAIC, a scalable multisensory enhancement of the torque proprioceptive AIF controller presented in [25] and the velocity controller presented in [22]. Our approach makes use of the alleged adaptability and robustness of AIF, taking advantage of previous works and overcoming some related limitations. We solved state estimation by combining representation learning and multimodal filtering with VFE optimization, improving the representational power and adaptability. Hence, we can perform online multisensory torque control, without the use of any dynamic or kinematic model of the robot at runtime. Furthermore, we performed a systematic comparison of several controllers on different experiments providing both qualitative and quantitative analysis on a robotic manipulator. The results showed that our proposed algorithm is more adaptive than state-of-the-art torque AIF baselines and classical controllers (MPC and IC), and it was more accurate in the presence of sensory noise, showing the strongest noise rejection capability. MAICs were highly adaptive and robust to different contexts, such as changes in the robot dynamics (i.e., elastic constraint) and changes in the robot properties (i.e., inertial properties). Furthermore, our simplified architecture makes the controller easy to deploy in any robotic manipulator. In line with the Bayesian hypothesis of how the brain processes the information from the senses, this work reinforces the idea that learning to predict can be directly transformed into adaptive control. The experimental validation shows the viability of this approach to standard industrial robotic tasks.

Appendix A
Model Predictive Controller

The results are compared to a standard model predictive torque control (MPC) formulation.

1) Optimization Problem: Neglecting external forces, the dynamics of the system are defined by the equation of motion as

$$\boldsymbol{\tau} = M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})\mathbf{q} + \mathbf{g}(\mathbf{q})$$

which is composed of the mass matrix M, the Coriolis matrix C, and the gravitational forces $\mathbf{g}$ [6]. Various approaches to compute the forward dynamics have been proposed [11]. The forward dynamics can be discretized to obtain the transition function

$$\mathbf{z}_{k+1} = f(\mathbf{z}_k, \mathbf{a}_k)$$

where $\mathbf{z}$ is the concatenated vector of joint positions, velocities, and accelerations.

The control problem can be formulated as an optimization problem as follows:

$$J^\star = \min_{\mathbf{z}0:N, \mathbf{a}_{0:N}} \sum_{k=0}^{N} J(\mathbf{z}_k, \mathbf{a}_k) \tag{32}$$

$$\text{s.t.} \quad \mathbf{z}_{k+1} = f(\mathbf{z}_k, \mathbf{a}_k) \tag{33}$$

$$\mathbf{a}_k \in \mathcal{U}, \mathbf{z}_k \in \mathcal{Z} \tag{34}$$

$$\mathbf{z}_0 = \mathbf{z}(0) \tag{35}$$

where J is the objective function, $\mathcal{U}$ and $\mathcal{Z}$ are the admissible sets of actions and states respectively, and $\mathbf{z}_0$ is the initial condition. The objective function was formulated as follows:

$$J(\mathbf{z}_k, \mathbf{a}_k) = \left(\mathbf{q}_k - \mathbf{q}_{\text{goal}}\right)^T W_{\text{goal}}\left(\mathbf{q}_k - \mathbf{q}_{\text{goal}}\right) + \mathbf{a}_k^T W_{\mathbf{a}}\mathbf{a}_k \tag{36}$$

TABLE III
PARAMETER SETTING FOR MPC

parameter	value
N	20
Δt	0.1s
W_{goal}	$400I_7$
$W_{\mathbf{a}}$	diag([1.75, 2, 2.5, 5, 20, 18.75, 62.5])

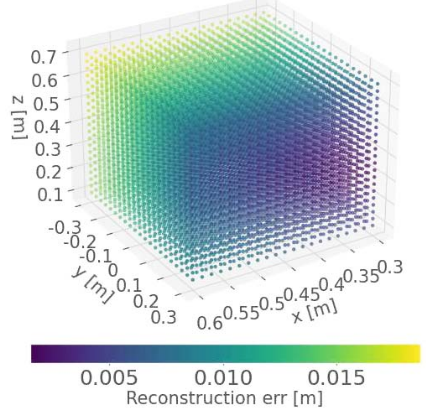


Fig. 8. End-effector reconstruction error.

where W_{goal} and W_τ are the weighting matrices for the goal configuration and the actions respectively.

2) Realization: In this work, we used the recursive Newton–Euler algorithm to solve the forward dynamics and a second-order explicit Runge–Kutta integrator. The parameter setting is summarized in Table III. In accordance to the time step, the control frequency is 10 Hz. The optimization problem is solved using the nonlinear solver proposed in [31] and the corresponding implementation [5]. The forward dynamics are computed using [11].

APPENDIX B
GP TRAINING

Fig. 8 illustrates a 3-D scatter plot that shows a heatmap of the end-effector reconstruction errors. Moreover, the axes define the cartesian workspace we considered in our experiments, where the robot base is placed at $\mathbf{x}_{\text{base}} = \{0, 0, 0\}$ and is frontally directed toward the x-direction. What is more, in order to define the training set we created a cubic grid of points over the defined workspace, splitting the cubic workspaces into 9261 points, 21 for every direction (i.e., x, y, and z axis). Consequently, we used an inverse kinematics algorithm from roboticstoolbox-python in order to define the joint values related to the obtained end-effector positions. We used 80% of these paired set as training set and the remaining 20% as the test set. Finally, from Fig. 8, it can be seen that on average the reconstruction error is roughly 0.010 m.

APPENDIX C
MULTIMODAL VAE TRAINING

In order to create the image data set, we used an impedance controller to explore the workspace defined in Appendix B and collect pictures of the robot in different poses. We used the joint values from the GP training set as a reference for the controller and with subscribers we collected both joints values and related images, creating a data set of 50 000 samples of paired joint values and images. The multimodal VAE was then trained using the loss function defined by (29). The network architecture and parameters are publicly available at https://github.com/Cmeo97/MAIC. Fig. 9 presents the average reconstruction loss during the training, and 50 epochs were used to train the network.

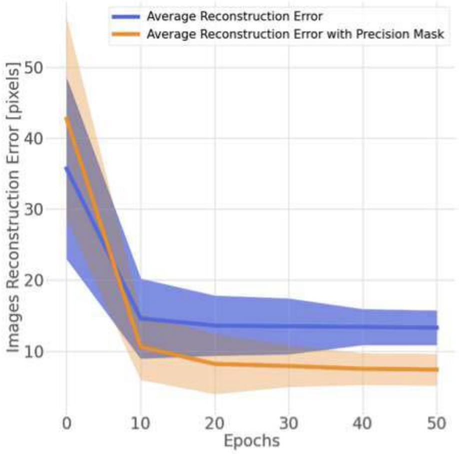


Fig. 9. Image reconstruction error.

APPENDIX D
MENTAL SIMULATION

Unlike most of the AIF controllers present in the literature, a great advantage of combining our approach with a multimodal VAE is the possibility to perform imagined simulations. In other words, given $\mathbf{x}_d$, the entire experiment can be simulated. Since sensory data are not available, the state update law becomes

$$\dot{\mathbf{z}} = -k_z \frac{\partial f}{\partial \mathbf{z}} \Sigma_f^{-1} (\mathbf{x}_d - f(\mathbf{z}, \boldsymbol{\rho})). \tag{37}$$

As a result, performing the integration step of the new internal state and decoding it, the updated $\{\mathbf{x_v}, \mathbf{x_q}\}$ can be computed and the new errors can be backpropagated again, creating a loop that allows the system to do imaginary simulations.

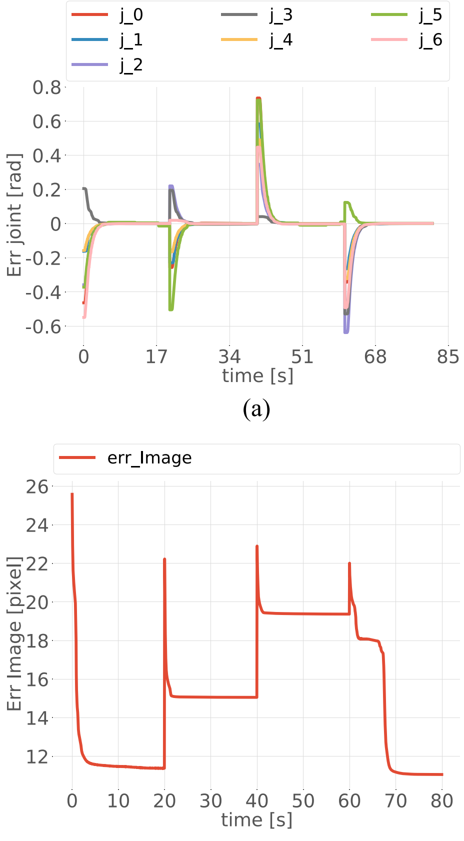
Fig. 10(a) and (b) shows, respectively, imagined joints error and images reconstruction error through the entire simulation. These results show that the errors converge faster to zero than in the normal regime [Fig. 2(a)] as it does not need to accommodate the real dynamics of the robot.

APPENDIX E
IMPEDANCE CONTROLLER

The presented impedance controller [10] is based on the following dynamic equation:

$$\boldsymbol{\tau} = K(\boldsymbol{q}_{\text{goal}} - \boldsymbol{q}) + D(-\dot{\boldsymbol{q}}) + C(\mathbf{q}, \dot{\mathbf{q}})\mathbf{q} + \mathbf{g}(\mathbf{q})$$

where K is the set joint stiffness, D is the corresponding critical damping, C is the Coriolis matrix, and g is the gravitational term. Considering that the dynamics of the robot are



(a)

(b)

Fig. 10. Mental simulation of sequential reaching of four goals. The goal is updated on time steps where peaks are present. (a) Joints errors of an imagined simulation. Each line represents the error of the *i*th joint. (b) Image reconstruction errors of an imagined simulation.

described by

$$M(\boldsymbol{q})\ddot{\boldsymbol{q}} + C(q, \dot{\boldsymbol{q}}) + g(g) = \boldsymbol{\tau} + \boldsymbol{\tau}_{\text{ext}} \tag{38}$$

with the impedance controller, the dynamics result in

$$M(\boldsymbol{q})\ddot{\boldsymbol{q}} = K(\boldsymbol{q}_{\text{goal}} - \boldsymbol{q}) + D(-\dot{\boldsymbol{q}}) + \boldsymbol{\tau}_{\text{ext}} \tag{39}$$

this translates in a second-order critically damped dynamics of the robot in the transition toward the desired goal.

REFERENCES

[1] M. Baioumy, P. Duckworth, B. Lacerda, and N. Hawes, "Active inference for integrated state-estimation, control, and learning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Xi'an, China, 2021, pp. 4665–4671.

[2] 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 *Proc. Eur. Control Conf. (ECC)*, 2021, pp. 1119–1125.

[3] C. L. Buckley, C. S. Kim, S. McGregor, and A. K. Seth, "The free energy principle for action and perception: A mathematical review," *J. Math. Psychol.*, vol. 81, pp. 55–79, Dec. 2017.

[4] A. Ciria, G. Schillaci, G. Pezzulo, V. V. Hafner, and B. Lara, "Predictive processing in cognitive robotics: A review," *Neural Comput.*, vol. 33, no. 5, pp. 1402–1432, 2021.

[5] A. Domahidi and J. Jerez. "Forces Professional." Embotech AG. 2019. [Online]. Available: https://embotech.com/FORCES-Pro

[6] R. Featherstone, *Rigid Body Dynamics Algorithms*. Boston, MA, USA: Springer, 2014.

[7] K. Friston, "The free-energy principle: A unified brain theory?" *Nat. Neurosci.*, vol. 11, no. 2, pp. 127–138, 2010.

[8] K. J. Friston, J. Daunizeau, J. Kilner, and S. J. Kiebel, "Action and behavior: A free-energy formulation," *Biol. Cybern.*, vol. 102, no. 3, pp. 227–260, 2010.

[9] K. J. Friston, N. Trujillo-Barreto, and J. Daunizeau, "DEM: A variational treatment of dynamic systems," *Neuroimage*, vol. 41, no. 3, pp. 849–885, 2008.

[10] N. Hogan, "Impedance control: An approach to manipulation: Part I—Theory," *J. Dyn. Syst. Meas. Control*, vol. 107, no. 1, pp. 1–7, 1985.

[11] L. M. G. Johannessen, M. H. Arbo, and J. T. Gravdahl, "Robot dynamics with URDF & CasADi," in *Proc. 7th (ICCMA)*, Delft, The Netherlands, 2019, pp. 185–190.

[12] M. Jung, T. Matsumoto, and J. Tani, "Goal-directed behavior under variational predictive coding: Dynamic organization of visual attention and working memory," in *Proc. IROS*, 2019, pp. 1040–1047.

[13] A. Koubaa, *Robot Operating System (ROS): The Complete Reference (Volume 2)*, 1st ed. Cham, Switzerland: Springer Publ. Company, Incorp., 2017.

[14] P. Lanillos and G. Cheng, "Active inference with function learning for robot body perception," in *Proc. Int. Workshop Continual Unsupervised Sensorimotor Learn.*, 2018, pp. 1–5.

[15] P. Lanillos and G. Cheng, "Adaptive robot body learning and estimation through predictive coding," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Madrid, Spain, 2018, pp. 4083–4090.

[16] P. Lanillos, S. Franklin, and D. W. Franklin, "The predictive brain in action: Involuntary actions reduce body prediction errors," *bioRxiv*, 2020. [Online]. Available: https://www.biorxiv.org/content/early/2020/07/08/2020.07.08.191304

[17] P. Lanillos, J. Pages, and G. Cheng, "Robot self/other distinction: Active inference meets neural networks learning in a mirror," in *Proc. 24th Eur. Conf. Artif. Intell. (ECAI)*, 2020, pp. 2410–2416.

[18] P. Lanillos and M. van Gerven, "Neuroscience-inspired perception-action in robotics: applying active inference for state estimation, control and self-perception," 2021, *arXiv:2105.04261*.

[19] T. Lesort, N. Díaz-Rodríguez, J.-F. Goudou, and D. Filliat, "State representation learning for control: An overview," *Neural Netw.*, vol. 108, pp. 379–392, Dec. 2018.

[20] C. Meo and P. Lanillos, "Multimodal VAE active inference controller," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2021, pp. 2693–2699.

[21] B. Millidge, A. Tschantz, A. K. Seth, and C. L. Buckley, "On the relationship between active inference and control as inference," in *Proc. Int. Workshop Active Inference*, 2020, pp. 3–11.

[22] G. Oliver, P. Lanillos, and G. Cheng, "An empirical study of active inference on a humanoid robot," *IEEE Trans. Cogn. Develop. Syst.*, early access, Jan. 8, 2021, doi: 10.1109/TCDS.2021.3049907.

[23] A. Paszke *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*. Red Hook, NY, USA: Curran Assoc., Inc., 2019, pp. 8024–8035.

[24] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, no. 85, pp. 2825–2830, 2011.

[25] C. Pezzato, R. Ferrari, and C. H. Corbato, "A novel adaptive controller for robot manipulators based on active inference," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 2973–2980, Apr. 2020.

[26] J. F. Queißer, B. Hammer, H. Ishihara, M. Asada, and J. J. Steil, "Skill memories for parameterized dynamic action primitives on the pneumatically driven humanoid robot child affetto," in *Proc. Joint IEEE 8th Int. Conf. Develop. Learn. Epigenet. Robot. (ICDL-EpiRob)*, Tokyo, Japan, 2018, pp. 39–45.

[27] J. F. Queisser, M. Jung, T. Matsumoto, and J. Tani, "Emergence of content-agnostic information processing by a robot using active inference, visual attention, working memory, and planning," *Neural Comput.*, vol. 33, no. 9, pp. 2353–2407, Aug. 2021.

[28] C. Sancaktar, M. A. J. van Gerven, and P. Lanillos, "End-to-end pixel-based deep active inference for body perception and action," in *Proc. Joint IEEE 10th Int. Conf. Develop. Learn. Epigenet. Robot. (ICDL-EpiRob)*, Valparaiso, Chile, 2020, pp. 1–8.

[29] S. Särkkä, *Bayesian Filtering and Smoothing*. Cambridge, U.K.: Cambridge Univ. Press, 2013.

[30] Y. Yamashita and J. Tani, "Emergence of functional hierarchy in a multiple timescale neural network model: A humanoid robot experiment," *PLoS Comput. Biol.*, vol. 4, no. 11, 2008, Art. no. e1000220.

[31] A. Zanelli, A. Domahidi, J. Jerez, and M. Morari, "FORCES NLP: An efficient implementation of interior-point methods for multistage nonlinear nonconvex programs," *Int. J. Control*, vol. 93, no. 1, pp. 13–29, 2020.