

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

Tim N. Nisslbeck¹^[0009-0007-3114-812X] and
Wouter M. Kouw¹^[0000-0002-0547-4817]

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

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

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

1 Introduction

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

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

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

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

2 Problem statement

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

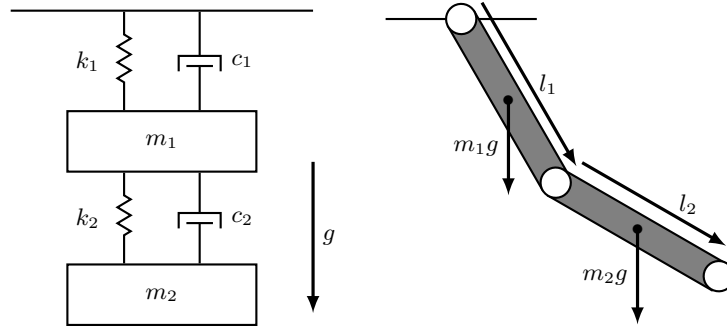


Fig. 1: (Left) A double mass-spring-damper system where block 1 is attached to a stationary frame and block 2 is attached to the first block. The dynamics of the system are determined by the masses m_i of the blocks, the stiffness of the springs k_i , the amount of friction c_i the dampeners provide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

3 Agent specification

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

3.1 Probabilistic model

We specify a likelihood function of the form:

$$p(y_k | \theta, \tau, u_k, \bar{u}_k, \bar{y}_k) = \mathcal{N}(y_k | \theta^\top [u_k \ \bar{u}_k \ \bar{y}_k], \tau^{-1}), \quad (1)$$

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

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

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

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

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

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

3.2 Inference

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

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

$$x_k = [u_k \ \bar{u}_k \ \bar{y}_k]. \quad (4)$$

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

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

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

$$p(\hat{y}_k | \hat{u}_k, \mathcal{D}_{k-1}) = \int p(\hat{y}_k | \theta, \tau, \hat{u}_k, \bar{u}_k, \bar{y}_k) p(\theta, \tau | \mathcal{D}_{k-1}) d(\theta, \tau). \quad (6)$$

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

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

where

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

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

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

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

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

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

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

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

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

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

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

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

$$p(y_t) \rightarrow p(y_t | y_*) \triangleq \mathcal{N}(y_t | m_*, v_*). \quad (15)$$

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

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

with a variational model of the form:

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

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

$$q^*(u_t) = \arg \min_{q \in Q} \mathcal{F}_k[q]. \quad (18)$$

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

$$\begin{aligned} \mathbb{E}_{q(y_t, u_t, \theta, \tau)} \left[\ln \frac{p(\theta, \tau \mid \mathcal{D}_k) q(u_t)}{p(\theta, \tau \mid y_t, u_t; \mathcal{D}_k) p(y_t \mid y_*) p(u_t)} \right] = & \quad (19) \\ \mathbb{E}_{q(u_t)} \left[\underbrace{\mathbb{E}_{p(y_t, \theta, \tau \mid u_t; \mathcal{D}_k)} \left[\ln \frac{p(\theta, \tau \mid \mathcal{D}_k)}{p(\theta, \tau \mid y_t, u_t; \mathcal{D}_k) p(y_t \mid y_*)} \right]}_{\mathcal{J}_k(u_t)} + \ln \frac{q(u_t)}{p(u_t)} \right]. \end{aligned}$$

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

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

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

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

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

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

$$p(y_t \mid u_t; \mathcal{D}_k) \triangleq \int p(y_t \mid \theta, \tau, u_t, \bar{u}_t, \bar{y}_t) p(\theta, \tau \mid \mathcal{D}_k) d(\theta, \tau) \quad (22)$$

$$= \mathcal{T}_{2\alpha_k} \left(y_t \mid \mu_k^\top x_t, \frac{\beta_k}{\alpha_k} (x_t^\top \Lambda_k^{-1} x_t + 1) \right), \quad (23)$$

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

$$\begin{aligned} \mathcal{J}_k(u_t) = \mathbb{E}_{p(y_t \mid u_t; \mathcal{D}_k)} \left[-\ln p(y_t \mid y_*) \right] & \quad (24) \\ - \mathbb{E}_{p(y_t, \theta, \tau \mid u_t; \mathcal{D}_k)} \left[\ln \frac{p(y_t, \theta, \tau \mid u_t; \mathcal{D}_k)}{p(\theta, \tau \mid \mathcal{D}_k) p(y_t \mid u_t; \mathcal{D}_k)} \right]. \end{aligned}$$

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

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

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

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

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

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

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

3.3 Coupling

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

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

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

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

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

3.4 Optimization

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

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

4 Experiments

4.1 System description

We perform an experiment¹ on a double mass-spring-damper system. Its equation of motions are the following second-order ordinary differential equations (ODE) [18]:

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

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

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

¹ Code found at <https://github.com/biaslab/IWAI2024-CARXEFE>

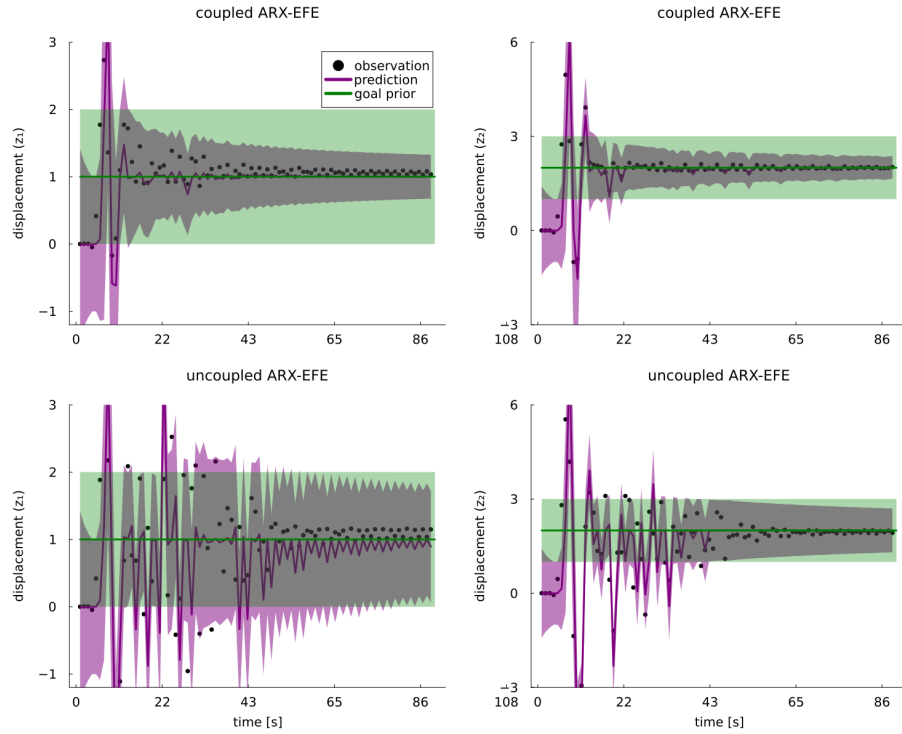
where \ddot{z}_t is calculated from the equations of motion in Eq. 30. The initial state of the system is the fixed point $z_0 = [0.0, 0.0, 0.0, 0.0]$. By reducing the step size Δt and correspondingly increasing the number of updates n_{iter} , we can reduce the risk of numerical instabilities. In our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

4.2 Comparisons

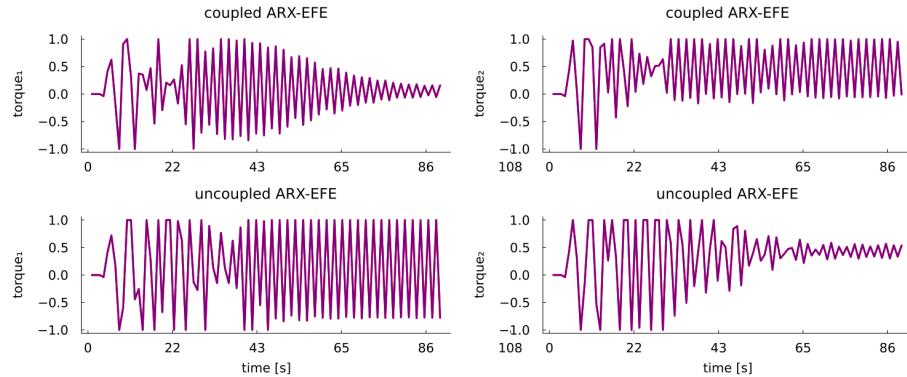
We compare a set of coupled ARX-EFE agents, referred to as CARX-EFE agents, with a set of uncoupled ARX-EFE agents. CARX-EFE and uncoupled ARX-EFE agents differ in the size M of the history vector x_k . For each buffer type, we use a history size of 2. Thus, an uncoupled ARX-EFE agent has a memory size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

4.3 Results

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



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



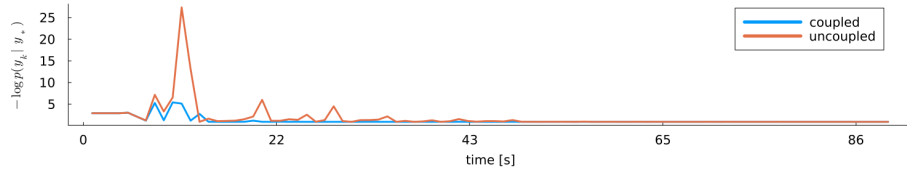
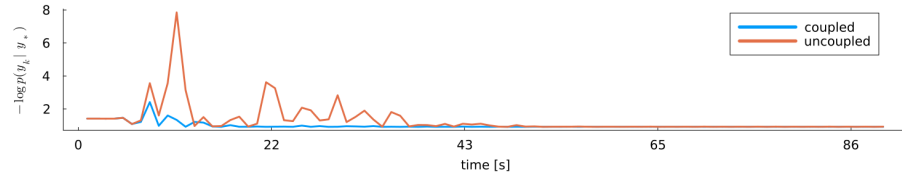
(b) Controls plotted over time. All agents exhibit a short inactivity phase in the beginning, before choosing non-zero controls. The control signals for both the coupled agent controlling mass m_1 and the uncoupled agent controlling mass m_2 have an initial period of large oscillations, which gradually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

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

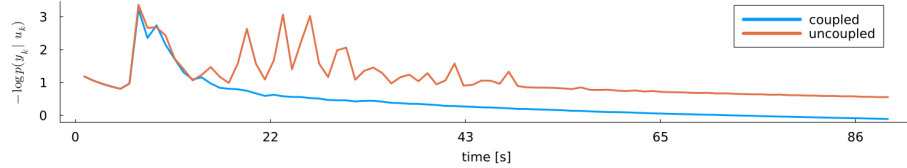
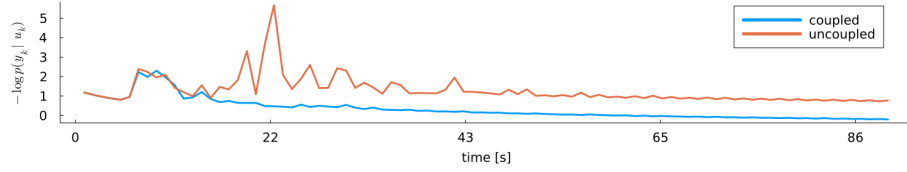
ARX-EFE agents oscillate more wildly (until around time step 45) and have more difficulty maintaining close adherence to the goal prior. They exhibit a prolonged oscillatory phase, where oscillations are more persistent and take significantly longer to dampen. The higher prediction variance further highlights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

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

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



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



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

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

5 Discussion

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

6 Conclusion

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

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

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

References

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

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

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