2025 IEEE 64th Conference on Decision and Control (CDC)
December 10-12, 2025. Rio de Janeiro, Brazil

eXplainable AI for data driven control: an inverse optimal control approach

Federico Porcari[1], Donatello Materassi[2], Simone Formentin[1]

2025 IEEE 64th Conference on Decision and Control (CDC) | 979-8-3315-2627-6/25/$31.00 ©2025 IEEE | DOI: 10.1109/CDC57313.2025.11312605

***Abstract*— Understanding the behavior of black-box data-driven controllers is a key challenge in modern control design. In this work, we propose an eXplainable AI (XAI) methodology based on Inverse Optimal Control (IOC) to obtain local explanations for the behavior of a controller operating around a given region. Specifically, we extract the weights assigned to tracking errors and control effort in the implicit cost function that a black-box controller is optimizing, offering a more transparent and interpretable representation of the controller's underlying objectives. This approach presents connections with well-established XAI techniques, such as Local Interpretable Model-agnostic Explanations (LIME) since it is still based on a local approximation of the control policy. However, the explanation provided by our method offers a structured and more control-relevant perspective. Numerical examples demonstrate that the inferred cost function consistently provides a deeper understanding of the controller's decision-making process.**

***Index Terms*— data-driven control, explainable AI, inverse optimal control**

I. Introduction

AI has been advancing rapidly, leading to its application across a wide array of domains, from healthcare and finance to manufacturing and transportation [1]–[4]. However, a major problem with AI-based algorithms is the lack of transparency in their decision-making processes [5], which has spurred the development of a new research field called eXplainable Artificial Intelligence (XAI), specifically focused on creating methods to interpret how these systems make decisions [6]. Despite the increasing interest in XAI methods, there is still no clear consensus or formal definition of what constitutes an "explanation." A common distinction is between *global explanations*, which aim to capture the overall behavior of the model across the entire input space, and *local explanations*, which focus on individual instances or trajectories, providing insights into why a particular outcome was produced in a given scenario [7].

Representative global explanation techniques include surrogate models such as decision trees [8], rule-based systems [9], and Accumulated Local Effects (ALE) [10]. Conversely, popular local explanation methods include LIME (Local Interpretable Model-Agnostic Explanations) [11], SHAP (SHapley Additive exPlanations) [12], and counterfactual reasoning [13]. Local explanations are often easier to compute, since they analyze only a restricted region of the input space rather than the full model. Despite this limitation, they remain valuable for debugging and failure analysis. Moreover, many local approaches can be categorized as *feature attribution methods*, where explanations are expressed as vectors that quantify the contribution of each input feature to the output [12].

Recently, AI techniques have also been increasingly applied to the design of control algorithms, with the goal of improving the performance of autonomous systems such as drones, self-driving cars, and robotic platforms [14]. The lack of transparency in AI-based control is particularly critical, since decisions are not made in isolation but within a feedback loop, often through constrained optimization over an extended horizon. These characteristics make control systems more complex to explain, motivating dedicated research on interpretable control [15], [16]. A key challenge is that most XAI methods are tailored to static tasks (e.g., classification), while control problems inherently involve temporal dynamics, where decisions evolve as time-series [17].

The main goal of this article is to develop a local explainability tool, that is specifically designed for control systems. Unlike standard XAI methods, which typically focus on estimating how much a single static feature influences the output, our approach aims to assess how individual signals affect the system's behavior over time. Our approach thus differs from previous XAI studies on dynamical systems [15], [16], as it explains the closed-loop interaction between plant and controller rather than focusing solely on the plant dynamics. Explanations of the controller itself have been considered only in [18], where SHAP was employed to analyze decision-making. In contrast, we propose a novel interpretability method that is rooted directly in control-theoretic principles, offering a more systematic perspective.

Given a black-box controller and a specific trajectory to explain, we derive a local explanation by linearizing the system around the trajectory of interest and then solving an inverse optimal control (IOC) problem. The resulting cost function serves as the explanation, since its weights quantify the relative importance of input and output signals in terms of control effort and performance. This reveals what the black-box controller is effectively optimizing and highlights the implicit trade-offs driving its decisions. More broadly, our explanation tool can be framed within a general

This work is partially supported by the FAIR project (NextGenerationEU, PNRR-PE-AI, M4C2, Investment 1.3), the 4DDS project (Italian Ministry of Enterprises and Made in Italy, grant F/310097/01-04/X56), and the PRIN PNRR project P2022NB77E (NextGenerationEU, CUP: D53D23016100001). It is also partly supported by the ENFIELD project (Horizon Europe, grant 101120657).

[1]Federico Porcari and Simone Formentin are with the Department of Electronics, Information and Bioengineering, Politecnico di Milano, via G. Ponzio 34/5, 20133 Milano, Italy. Email: {federico.porcari, simone.formentin}@polimi.it

[2]Donatello Materassi is with Department of Electrical and Computer Engineering, University of Minnesota, Twin Cities. Email: mater013@umn.edu

perspective that enables a formal definition of explanations. In this view, an explanation is interpreted as a *reduced-complexity model* [19], from which interpretable information can be systematically extracted. While many XAI methods rely on some underlying simplified model, our framework explicitly introduces an *interpretability function* to formalize how explanations are constructed and evaluated. This function specifies the metrics to be extracted from the reduced model, ensuring rigor and consistency rather than ad hoc choices. In this article, the interpretability function outputs the cost weights obtained through IOC, but we also discuss how alternative choices could align our framework with existing XAI methods, thereby broadening its applicability.

The remainder of the paper is as follows. Section II formally states the problem under consideration. In Section III, we introduce interpretability functions for LTI systems, highlighting the role of inverse optimal control in uncovering the trade-offs implicit in a black-box control policy. Finally, Section IV presents numerical examples showing how the proposed approach reveals non-intuitive behaviors in controlled systems.

II. Problem Statement

We consider a discrete-time, time-invariant dynamical system described by:

$$x(k+1) = f(x(k), u(k)), \tag{1a}$$
$$y(k) = g(x(k), u(k)), \tag{1b}$$

where f and g are unknown functions governing the system dynamics. Here, $x(k)$ represents the system state at time step k, while $u(k)$ and $y(k)$ denote the input and output signals, respectively. We assume that $u(k)$ is a controllable input, and we have access to experimental data in the form of input-output (I/O) pairs:

$$\mathcal{D}_{N_{data}} = \{(u(k), y(k))\}_{k=1}^{N_{data}}, \tag{2}$$

where N_{data} is the number of collected data points.

The goal of data-driven control is to directly synthesize a control law from the dataset $\mathcal{D}_{N_{data}}$, bypassing explicit identification of the system dynamics in (1). Specifically, we seek a control law of the form:

$$u(\cdot) = \phi(y(\cdot)), \tag{3}$$

where ϕ is an operator mapping the measured output signal into the control signal to achieve a desired closed-loop behavior. Several approaches exist to achieve this without first identifying a model of the system. Prominent methodologies include:

- *Reinforcement Learning (RL)*: RL-based strategies learn an optimal control policy by interacting with the system and optimizing a performance criterion. Techniques such as *Q-learning* [20] and *policy gradient* methods [21] have demonstrated effectiveness in high-dimensional and nonlinear settings, although they often require extensive exploration and suffer from sample inefficiency.

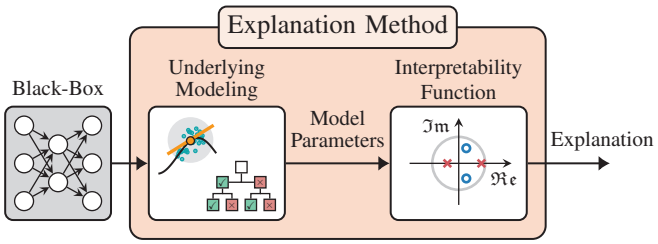


Fig. 1: *Schematic formal representation of an explanation method.*

- *Willems' Fundamental Lemma*: Originally formulated for linear time-invariant (LTI) systems [22], this method enables direct control design using raw data. Recent works have extended this concept to certain nonlinear systems [23].
- *Model Reference Approaches*: These methods, often used in adaptive control [24], aim to shape the system's behavior to match a desired reference model through data-driven techniques [25]–[27]. Some theoretical studies also compare direct data-driven approaches with traditional system identification, identifying conditions under which the former is preferable [28].
- *Data-Driven Predictive Control*: Methods based on subspace identification [29] use past data to predict future system behavior and compute control actions accordingly [30], [31]. Extensions incorporating, e.g., Gaussian process regression [32] further enhance predictive capabilities for nonlinear systems.

Although this list is not exhaustive, it highlights the increasing adoption of data-driven control techniques in real-world applications. As these methods become more prevalent, particularly in safety-critical domains such as autonomous vehicles or healthcare, understanding the reasoning behind a controller's decisions becomes crucial.

In this work, we tackle the explainability problem for data-driven controllers following a two-step approach, as illustrated in Figure 1. The first step involves selecting an underlying explanatory model from an interpretable class of models. This model is not required to capture the full complexity of the system, but should instead provide a simplified yet insightful representation. Indeed, the choice of the model class should be guided by interpretability as the primary criterion. Once this modeling procedure is established, the second step involves deriving an explanation by mapping the parameters of the simplified model into quantitative measures that assess the internal mechanisms of the original complex system. This mapping is formalized through what we refer to as the "interpretability function".

Many XAI methods, though not always described this way, implicitly follow a two-step structure. For instance, LIME uses linear regression as its interpretable model and reports the regression coefficients as explanations, effectively applying the identity function as its interpretability function [15]. Similarly, SHAP constructs explanations through linear

models and averages feature contributions across all $n!$ permutations of the inputs, thereby employing an interpretability function defined as this averaging process [33].

Recognizing this common structure helps formalize what it means to explain a black-box model and supports systematic comparisons across methods. Our work, however, focuses specifically on feedback controllers, where explanations must capture the closed-loop interaction between plant and controller. Standard XAI techniques often fail in this regard, as they are designed for static tasks and cannot account for the temporal dependencies and feedback inherent in control problems.

In the next section, we motivate our choices for both the underlying model class and the interpretability function, showing how they address the unique challenges of explaining data-driven controllers in dynamic systems.

III. Inverse optimal control as an interpretability function

In this section, we propose a specific tool for the explainability of feedback controllers. Following the approach delineated in the previous section, we need to select a class of interpretable models and an interpretability function that can translate the parameters of these models into meaningful quantities capable of representing the behavior of the controller within the feedback loop. In this respect, we choose

- Linear Time-Invariant (LTI) systems as our class of interpretable models
- the solution of an inverse Linear Quadratic Regulator (LQR) based on the linearized dynamics of the controller as our interpretable function.

and motivate these choices in the following.

A. LTI systems as class of interpretable models

While nonlinear systems exhibit complex dynamics, meaningful insights can still be gained by analyzing interpretable approximations of their local behavior. A natural choice for the class of interpretable models is given by Linear Time-Invariant (LTI) systems. They are mathematically tractable, with a rich theoretical foundation that enables precise analysis of system properties such as stability, controllability, and observability. Furthemore, using LTI models ensures consistency with existing explainable AI methods, many of which rely on linear approximations (e.g., LIME), making the application of these methods to dynamic systems more coherent and effective.

B. Interpretability functions for LTI systems

The choice of linearization of the controller's dynamics as the underlying modeling procedure enables the selection of several natural interpretability functions, which include:

- *Difference Equation Coefficients.* Given the linearized behavior of the controller, the interpretability function can be chosen to provide the coefficients of the numerator and denominator of the corresponding transfer function, offering insights into the system's input-output relationships and its response to control inputs. These coefficients also provide a natural way to describe the intertemporal relationships among the different time signals of the system.
- *Pole-Zero Representation.* An alternative (and equivalent) interpretability function could be given by the system's poles and zeros. Poles provide critical information about stability and transient response, while zeros influence controllability and system performance. Examining how these characteristics change under different conditions can offer valuable insights into the controller's impact on system behavior especially in terms of time constants and time separations.
- *Inverse Optimal Control.* A more nuanced yet powerful interpretability function involves the use of inverse optimal control techniques, to infer a cost function that explains the controller's local behavior. This approach enables a *structured interpretation of control strategies* in terms of trade-offs between output error and control effort. Moreover, it provides a systematic way to analyze controllers that were tuned empirically or derived from learning-based methods, revealing hidden objectives that may not have been explicitly defined.

The first two interpretability functions are equivalent, differing only by a direct transformation. They naturally extend existing XAI techniques to the dynamic setting, offering a straightforward way to analyze input–output relationships and stability properties of the controller. In contrast, the inverse optimal control (IOC) approach is inherently control-specific and directly tied to performance, as it reveals how signals interact according to the controller's objectives and priorities. Moreover, IOC can be extended beyond linear systems, making it a versatile tool for explaining feedback controllers in diverse applications. For these reasons, this article adopts IOC as the interpretability function of choice.

C. Inverse LQR for explainability

Since the interpretable class of models is given by LTI systems, the most natural approach to solving an inverse optimal control problem is by considering the standard formulation of inverse Linear Quadratic Regulator (LQR) optimal control. The conditions for the existence of a solution to such a problem in the case of linear dynamics and a stabilizing controller are quite general [34] guaranteeing the existence of the explanation. The mathematical details of the numerical methods employed in this paper to compute the inverse LQR problem are inspired from [35] and described in the following.

The standard finite-horizon LQR problem is given by

$$\min_{x,u} x(N)^\top S x(N) + \sum_{k=1}^{N-1} (u(k)^\top R u(k) + x(k)^\top Q x(k)) \tag{4}$$

$$\text{s.t. } x(k+1) = Ax(k) + Bu(k), \quad x(1) = \bar{x}, \tag{5}$$

where $S, Q \succeq 0$, and $R \succ 0$. Such inverse optimal control problem seeks to determine (S, Q, R) given (A, B) and observed optimal trajectories or control inputs, potentially

with noise[1]. In this work, we assume $S = 0$ for simplicity. Nonetheless, the problem remains well-posed as Q is uniquely determined for a given closed-loop system [35].

Assume to have access to M sets of state-input trajectories $\{\mathcal{D}_N^{(i)}\}_{i=1}^M$ generated by the data-driven controller[2], namely $\mathcal{D}_N^{(i)} = \{(x_d^{(i)}(k), u_d^{(i)}(k))\}_{k=1}^N$, where the subscript d indicates that data is affected by noise. To explain each collected closed-loop trajectory, we leverage the Pontryagin Maximum Principle (PMP), which provides necessary and sufficient conditions for optimality through the existence of an adjoint variable λ that satisfies the equations:

$$\lambda(k) = A^\top \lambda(k+1) + Qx(k), \quad k = 2, \dots, N-1, \tag{6a}$$
$$\lambda(N) = 0, \tag{6b}$$
$$u(k) = -R^{-1}B^\top \lambda(k+1), \quad k = 1, \dots, N-1. \tag{6c}$$

Using these conditions, we can search for the matrices Q and R that satisfy the PMP given the collected datasets $\mathcal{D}_N^{(i)}$. Since each dataset is noisy, we cannot use the measured state $x_d^{(i)}$ to solve (6). Instead, we estimate the noiseless state sequence by minimizing the difference between the predicted evolution $x^{(i)}$ and the measured trajectories $x_d^{(i)}$. To this end, considering the state equation (5) and the PMP input equation (6c), we write the noiseless state dynamics as

$$x(k+1) = Ax(k) - BR^{-1}B^\top \lambda(k+1). \tag{7}$$

Through (7), the state dynamics can be reconstructed from an initial condition $\bar{x}$, the cost function matrices Q, R, and the adjoint variable update (6a). Therefore, we can obtain estimates $\hat{Q}$, $\hat{R}$ of the cost function matrices by taking the minimum state reconstruction error over each collected data, namely solving the optimization problem

$$\min_{Q,R} \quad \frac{1}{M}\sum_{i=1}^{M}\sum_{k=2}^{N} \left\| x_d^{(i)}(k) - x^{(i)}(k) \right\|^2 \tag{8a}$$
$$\text{s.t.} \quad \lambda^{(i)}(k) = A^\top \lambda^{(i)}(k+1) + Qx^{(i)}(k), \tag{8b}$$
$$x^{(i)}(k+1) = Ax^{(i)}(k) - BR^{-1}B^\top \lambda^{(i)}(k+1), \tag{8c}$$
$$\lambda^{(i)}(N) = 0, \tag{8d}$$
$$x^{(i)}(1) = \bar{x}^{(i)}, \tag{8e}$$

which is convex and can be solved using standard nonlinear optimization solvers. Moreover, for $M \to \infty$, the estimates $\hat{Q}$, $\hat{R}$ of the IOC problem (8) converge to the true values of the forward problem (4) [35].

IV. Numerical examples

In this section, we illustrate the IOC explainability framework through two case studies: a second-order LTI system and an inverted pendulum. The trajectories required for solving the IOC problem (8) are obtained by simulating the systems in closed loop with a model predictive controller (MPC). Although an MPC is not strictly a data-driven controller, we treat the resulting I/O trajectories as originating from an unknown control strategy to be explained. This setup enables a direct comparison between the inferred explanations $(\hat{Q}, \hat{R})$ and the ground-truth weights (Q_{MPC}, R_{MPC}) used to tune the MPC.

A. Second-order LTI system

Consider the fully measurable second-order system

$$x(k+1) = Ax(k) + Bu(k),$$
$$y(k) = x(k) + e(k),$$

where e is a zero-mean white measurement noise with covariance $\sigma^2 I$, $\sigma = 0.01$, and

$$A = \begin{bmatrix} 1 & 1 \\ -0.5 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}.$$

The system is controlled using an unconstrained quadratic MPC that regulates the state to zero with the cost function weights matrices

$$Q_{MPC} = \begin{bmatrix} 0.3 & 0 \\ 0 & 0.1 \end{bmatrix}, \quad R_{MPC} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Our objective is to explain the behavior of the MPC for different prediction horizons T. To this end, we conduct two experiments with $T = 5$ and $T = 10$. In each experiment, we collect $M = 30$ closed-loop trajectories of length $N = 30$ by sampling the initial conditions $x(0)$ from a multivariate uniform distribution over $[-2, 2] \times [-2, 2]$. Then, we compute an equivalent LQR-based explanation $(\hat{Q}, \hat{R})$ through the IOC algorithm in (8), yielding the following matrices:

$$\hat{Q}_{T=10} = \begin{bmatrix} 0.265 & 0.015 \\ 0.015 & 0.112 \end{bmatrix}, \quad \hat{R}_{T=10} = \begin{bmatrix} 0.988 & 0 \\ 0 & 1.012 \end{bmatrix},$$
$$\hat{Q}_{T=5} = \begin{bmatrix} 0.014 & -0.02 \\ -0.02 & 0.041 \end{bmatrix}, \quad \hat{R}_{T=5} = \begin{bmatrix} 1.06 & 0 \\ 0 & 0.949 \end{bmatrix}.$$

For $T = 10$, the obtained explanation resembles the original MPC weights; however, for $T = 5$, the explanation matrices show a different trend. In particular, the magnitude of $\hat{Q}_{T=5}$ compared to $\hat{R}_{T=5}$ is smaller, indicating a less aggressive control strategy compared to $T = 10$ (see Figure 2 for a comparison between $T = 5$ and $T = 10$).

Moreover, while Q_{MPC} assigns a larger penalty to the first state, the explanation $\hat{Q}_{T=5}$ weights more the second state. This result indicates that, *over the full simulation length* $N = 30$, the LQR that best approximates the measured closed-loop behavior is characterized by weight matrices which differ in nature from than those used in the original MPC formulation.

To validate this claim, Figure 2 shows the MPC trajectories compared to the LQR trajectories computed both using Q_{MPC}, R_{MPC} and $\hat{Q}, \hat{R}$. When $T = 5$, the trajectories obtained with $\hat{Q}, \hat{R}$ match more closely the MPC trajectories.

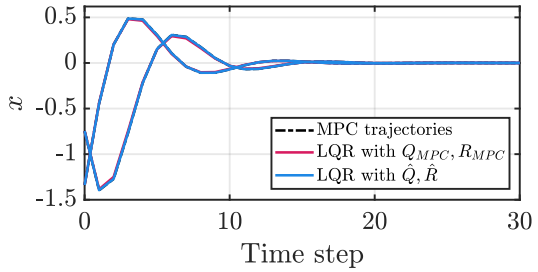
B. Inverted pendulum

Next, we consider the fully measurable inverted pendulum [36] with dynamics

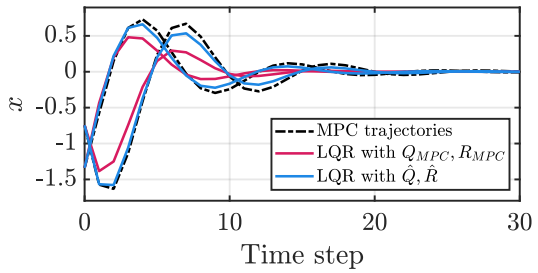
$$\theta(k+1) = \theta(k) + \tau\omega(k),$$
$$\omega(k+1) = \frac{\tau g}{l}\sin(\theta(k)) + \left(1 - \frac{\tau d}{ml^2}\right)\omega(k) + \frac{\tau}{ml^2}u(k),$$

[1]If the system matrices are not available, they can be estimated from the same data using identification techniques, such as subspace methods.

[2]These datasets can be obtained by simulating the system in closed-loop.



(a) $T = 10$.



(b) $T = 5$.

Fig. 2: *LQR trajectories using MPC weights Q_{MPC}, R_{MPC} and using IOC weights $\hat{Q}$, $\hat{R}$.*

Parameter [unit]	m [kg]	g [m/s^2]	l [m]	d [Nms]	τ [s]
Value	0.676	9.81	0.45	0.1	0.02

TABLE I: Parameters of the inverted pendulum.

where θ and ω are the angular position and speed, respectively, and u is the control input. The system's parameters are listed in Table I.

Since the pendulum dynamics are nonlinear and the IOC algorithm assumes an underlying linear system, we focus on explaining the local behavior of the pendulum around a given angular position $\bar{\theta}$. To do so, we collect $M = 30$ closed-loop trajectories of length $N = 50$ in the neighborhood of $\bar{\theta}$ using an unconstrained nonlinear MPC (NMPC) with prediction horizon $T = 5$. The NMPC cost function is quadratic and penalizes the deviation of $\theta(k)$ and $u(k)$ from the reference values $\theta_r(k)$, $u_r(k)$, with $\theta_r(k) = \bar{\theta} + e(k)$, where $e \sim \mathcal{N}(0, 0.05^2)$ is a Gaussian perturbation that excites the system, and $u_r(k)$ is the equilibrium input corresponding to $\theta_r(k)$. The NMPC cost function matrices are

$$Q_{MPC} = \begin{bmatrix} 1000 & 0 \\ 0 & 100 \end{bmatrix}, \quad R_{MPC} = 1,$$

and the initial conditions $\theta(0)$, $\omega(0)$ are sampled from a multivariate uniform distribution over $[\bar{\theta} - 0.3, \bar{\theta} + 0.3] \times [-0.3, 0.3]$.

With the measured trajectories, we compute a least-squares estimate of the linearized system matrices $A(\bar{\theta}), B(\bar{\theta})$ and then solve the IOC problem. Since the pendulum dynamics are nonlinear, the behavior of the NMPC, and consequently its explanation, depends on the operating point of the system. For this reason, we study the explanations for the two different angular positions $\bar{\theta}_1 = 0$ rad (upright position) and $\bar{\theta}_2 = \pi/2$ rad (horizontal position).

The explanations obtained by the IOC algorithm for the two angles $\bar{\theta}_1$, $\bar{\theta}_2$ are

$$\hat{Q}_{\bar{\theta}_1} = \begin{bmatrix} 11.674 & 18.179 \\ 18.179 & 28.309 \end{bmatrix}, \quad \hat{Q}_{\bar{\theta}_2} = \begin{bmatrix} 12.949 & 12.408 \\ 12.408 & 19.183 \end{bmatrix},$$

where $\hat{R}_{\bar{\theta}_1} = \hat{R}_{\bar{\theta}_2} = R_{MPC} = 1$ is forced by the optimization problem[3]. Comparing $\hat{Q}_{\bar{\theta}_1}$ and $\hat{Q}_{\bar{\theta}_2}$, we observe that the weights are larger for the operating point $\bar{\theta}_1 = 0$ rad. Since the IOC computes the optimal ratio between the weight matrices Q and R, larger weights in $\hat{Q}_{\bar{\theta}_1}$ imply that the NMPC penalizes the control input less when the pendulum is upright. This behavior is coherent with a physical understanding of the pendulum dynamics, as a larger input is required to keep the pendulum horizontal. We observe that, in this case, the matrices providing the explanation exhibit non-negligible off-diagonal components, indicating that the controller is optimizing a cost function where the state components are significantly coupled. This contrasts with the nominal cost function used in the MPC, which is diagonal, suggesting that the learned controller accounts for interactions between different state variables rather than treating them independently.

In order to validate this point, we can attempt to solve the optimization problem (8) while enforcing the matrices to be diagonal, leading to

$$\hat{Q}_{\bar{\theta}_1,d} = \begin{bmatrix} 63.901 & 0 \\ 0 & 49.001 \end{bmatrix}, \quad \hat{Q}_{\bar{\theta}_2,d} = \begin{bmatrix} 29.125 & 0 \\ 0 & 19.578 \end{bmatrix}.$$

Figure 3 shows a comparison between the LQR trajectories computed using the MPC weight Q_{MPC}, the non-diagonal explanations $\hat{Q}$ and the diagonal explanation $\hat{Q}_d$ for $\bar{\theta}_1$, highlighting that the adoption of a non-diagonal $\hat{Q}$ maximizes adherence to the measured trajectories, especially in transient, and hence provides a more accurate description of the actual cost being optimized by the controller.

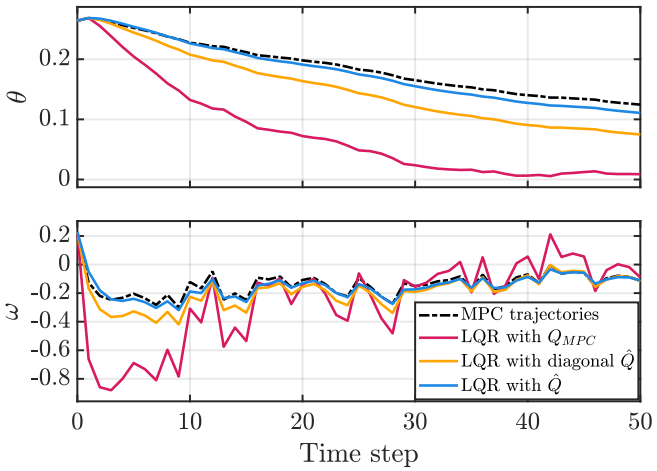


Fig. 3: *MPC and LQR trajectories for the pendulum near the equilibrium point $\bar{\theta}_1 = 0$ rad.*

[3]If $\hat{Q}$, $\hat{R}$ are solutions of (8), then, for any $\alpha > 0$, also $\alpha\hat{Q}$, $\alpha\hat{R}$ are solutions of (8). Therefore, we impose $\hat{R} = 1$ to obtain a unique solution of the IOC.

V. Concluding remarks

Developing explainability tools for AI-based and data-driven controllers is crucial for transparency, debugging, validation, and regulatory compliance. Yet, dynamic controllers pose specific challenges due to temporal dependencies and closed-loop interactions.

To address this, we propose a methodology that combines control-theoretic tools with interpretability functions. The framework builds explanations through a *reduced-complexity model* (obtained by linearizing the system around the trajectory of interest) and an *interpretability function* derived from the cost weights of a linear quadratic problem. Illustrative examples show how this approach clarifies counterintuitive behaviors and exposes unintended effects in data-driven control design.

Future directions include extending the method to nonlinear systems, improving scalability for real-time and high-dimensional settings, and incorporating domain knowledge.

References

[1] J. Chen, K. Li, Z. Zhang, K. Li, and P. S. Yu, "A survey on applications of artificial intelligence in fighting against COVID-19," *ACM Computing Surveys (CSUR)*, vol. 54, no. 8, pp. 1–32, 2021.

[2] X. Li, A. Sigov, L. Ratkin, L. A. Ivanov, and L. Li, "Artificial intelligence applications in finance: a survey," *Journal of Management Analytics*, vol. 10, no. 4, pp. 676–692, 2023.

[3] S. Bi, C. Wang, B. Wu, S. Hu, W. Huang, W. Ni, Y. Gong, and X. Wang, "A comprehensive survey on applications of AI technologies to failure analysis of industrial systems," *Engineering Failure Analysis*, p. 107172, 2023.

[4] P. Hamet and J. Tremblay, "Artificial intelligence in medicine," *Metabolism*, vol. 69, pp. S36–S40, 2017.

[5] M. Ryan, "In AI we trust: ethics, artificial intelligence, and reliability," *Science and Engineering Ethics*, vol. 26, no. 5, pp. 2749–2767, 2020.

[6] R. Dwivedi, D. Dave, H. Naik, S. Singhal, R. Omer, P. Patel, B. Qian, Z. Wen, T. Shah, G. Morgan *et al.*, "Explainable AI (XAI): Core ideas, techniques, and solutions," *ACM Computing Surveys*, vol. 55, no. 9, pp. 1–33, 2023.

[7] D. Minh, H. X. Wang, Y. F. Li, and T. N. Nguyen, "Explainable artificial intelligence: a comprehensive review," *Artificial Intelligence Review*, pp. 1–66, 2022.

[8] A. Sivaprasad, E. Reiter, N. Tintarev, and N. Oren, "Evaluation of human-understandability of global model explanations using decision tree," in *European Conference on Artificial Intelligence*. Springer, 2023, pp. 43–65.

[9] J. van der Waa, E. Nieuwburg, A. Cremers, and M. Neerincx, "Evaluating xai: A comparison of rule-based and example-based explanations," *Artificial intelligence*, vol. 291, p. 103404, 2021.

[10] D. W. Apley and J. Zhu, "Visualizing the effects of predictor variables in black box supervised learning models," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 82, no. 4, pp. 1059–1086, 2020.

[11] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should I trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 1135–1144.

[12] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30, 2017.

[13] R. K. Mothilal, A. Sharma, and C. Tan, "Explaining machine learning classifiers through diverse counterfactual explanations," in *Proceedings of the 2020 conference on fairness, accountability, and transparency*, 2020, pp. 607–617.

[14] C. Zhang, J. Wang, G. G. Yen, C. Zhao, Q. Sun, Y. Tang, F. Qian, and J. Kurths, "When autonomous systems meet accuracy and transferability through AI: A survey," *Patterns*, vol. 1, no. 4, 2020.

[15] D. Biparva and D. Materassi, "Application of explainable AI and causal inference methods to estimation algorithms in networks of dynamic systems," in *2023 American Control Conference (ACC)*. IEEE, 2023, pp. 1889–1894.

[16] G. Riva and S. Formentin, "Towards explainable data-driven control (XDDC): the property-preserving framework." *IEEE Control Systems Letters*, 2024.

[17] D. Materassi, S. Warnick, K. Vemuru, F. Vatan, P. L. Petrillo, A. Kamara, and B. Henz, "Explaining complex systems: a tutorial on transparency and interpretability in machine learning models," in *2024 IFAC Symposium on System Identification*. IFAC, 2024.

[18] J. P. Allamaa, P. Patrinos, and T. D. Son, "Exampc: the data-driven explainable and approximate nmpc with physical insights," 2025. [Online]. Available: https://arxiv.org/abs/2503.00654

[19] D. Materassi, G. Innocenti, and L. Giarré, "Reduced complexity models in the identification of dynamical networks: links with sparsification problems," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*. IEEE, 2009, pp. 4796–4801.

[20] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. MIT Press, 2018.

[21] B. Recht, "A tour of reinforcement learning: The view from continuous control," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 2, pp. 253–279, 2019.

[22] J. C. Willems, "A note on persistency of excitation," *Systems & Control Letters*, vol. 54, no. 4, pp. 325–329, 2005.

[23] O. Molodchyk and T. Faulwasser, "Exploring the links between the fundamental lemma and kernel regression," *IEEE Control Systems Letters*, 2024.

[24] K. S. Narendra and A. M. Annaswamy, *Stable Adaptive Systems*. Prentice-Hall, 1989.

[25] V. Breschi, C. De Persis, S. Formentin, and P. Tesi, "Direct data-driven model-reference control with lyapunov stability guarantees," in *2021 60th IEEE conference on decision and control (CDC)*. IEEE, 2021, pp. 1456–1461.

[26] T. O. de Jong, V. Breschi, M. Schoukens, and S. Formentin, "Data-driven model-reference control with closed-loop stability: the output-feedback case," *IEEE Control Systems Letters*, vol. 7, pp. 2431–2436, 2023.

[27] R. Busetto, V. Breschi, and S. Formentin, "Meta-learning for model-reference data-driven control," *Automatica*, vol. 172, p. 112006, 2025.

[28] S. Formentin, K. Van Heusden, and A. Karimi, "A comparison of model-based and data-driven controller tuning," *International Journal of Adaptive Control and Signal Processing*, vol. 28, no. 10, pp. 882–897, 2014.

[29] Z. Hou and Z. Wang, "From model-based control to data-driven control: Survey, classification and perspective," *Information Sciences*, vol. 235, pp. 3–35, 2013.

[30] V. Breschi, A. Chiuso, and S. Formentin, "Data-driven predictive control in a stochastic setting: A unified framework," *Automatica*, vol. 152, p. 110961, 2023.

[31] A. Chiuso, M. Fabris, V. Breschi, and S. Formentin, "Harnessing uncertainty for a separation principle in direct data-driven predictive control," *Automatica*, vol. 173, p. 112070, 2025.

[32] L. Hewing, K. Wabersich, M. Menner, and M. Zeilinger, "Learning-based model predictive control: Toward safe learning in control," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 3, pp. 269–296, 2020.

[33] D. Biparva and D. Materassi, "Incorporating information into shapley values: Reweighting via a maximum entropy approach," in *Internation Conference on Machine Learning*. Elsevier, 2024.

[34] M. C. Priess, R. Conway, J. Choi, J. M. Popovich, and C. Radcliffe, "Solutions to the inverse lqr problem with application to biological systems analysis," *IEEE Transactions on control systems technology*, vol. 23, no. 2, pp. 770–777, 2014.

[35] H. Zhang, J. Umenberger, and X. Hu, "Inverse optimal control for discrete-time finite-horizon linear quadratic regulators," *Automatica*, vol. 110, p. 108593, 2019.

[36] L. Gross, L. Maywald, S. Kumar, F. Kirchner, and C. Lüth, "Analytic estimation of region of attraction of an LQR controller for torque limited simple pendulum," in *2022 IEEE 61st Conference on Decision and Control (CDC)*, 2022, pp. 2695–2701.