IEEE Control Systems Letters paper presented at
2024 IEEE 63rd Conference on Decision and Control (CDC)
December 16-19, 2024. MiCo, Milan, Italy

Towards eXplainabile Data-Driven Control (XDDC): the property-preserving framework

Giorgio Riva, Simone Formentin

***Abstract*— As Artificial Intelligence (AI) techniques continue to advance, the need for explainability becomes increasingly crucial, especially in sensitive or safety-critical domains. eXplainable AI (XAI) has emerged to address this need, aiming to enhance transparency in complex models. While XAI has gained traction in mainstream machine learning, its application in data-driven control systems remains relatively unexplored. This paper introduces a novel concept of explainability tailored for data-driven control, allowing one to design feedback loops from data incorporating prior knowledge and preserving important system properties. Through two case studies, we demonstrate the efficacy of this property-preserving framework in direct and indirect data-driven control system design. This work lays the foundation for further research at the intersection of AI and data-driven control, offering insights into enhancing transparency in complex control systems.**

***Index Terms*— Data driven control, Machine Learning, Identification for control.**

I. Introduction

In the past decade, AI and ML techniques have rapidly expanded across various fields such as medicine [1], finance [2], and transportation [3], [4], resulting in significant accuracy improvements. However, this growth has also led to the development of more complex and less interpretable models [5]. This lack of transparency has highlighted the importance of explainability in AI, particularly in sensitive domains like medicine, where it is considered a crucial performance metric alongside accuracy for real-world deployment [6]. Consequently, eXplainable Artificial Intelligence (XAI) has gained popularity, with efforts focused on developing tools to enhance model transparency. These tools aim to improve user understanding of the decision-making process and potentially mitigate biases in the dataset [5], [6], [7].

In the XAI community, explainability focuses on two main aspects: the dataset and the model. Pre-modelling explainability, discussed in [7], involves analyzing and processing the dataset before model training through data analysis, data summarization, and feature engineering [5]. Regarding models, explanations vary depending on the model class, as outlined in [6]. They can be global or local, direct or via surrogates, and specific to the model or agnostic. White-box models like linear models and decision trees offer ante-hoc explainability, i.e., by design, with dimensionality and sparsity being crucial factors in enhancing explainability [6], [5]. On the other hand, black-box model explanations are retrieved post-hoc, i.e. after the actual training, with both global and local explanations achievable through surrogate models like SHAP and LIME [8], [9]. In summary, XAI in machine learning aims to balance prediction performance and model complexity, prioritizing understanding alongside accuracy.

Explainability is becoming a hot research topic also in the control community, where many current data-driven design approaches borrow techniques from standard ML. However, as far as we are aware, only few works deal with this problem: in [10] a symbolic regression model is learnt, combining math operations and variables via a genetic programming algorithm; in [11] and [12], instead, a decision tree is used to approximate big complex controllers stored via big look-up tables; while in [13], a qualitative description of the controller is built from simulations. In the field of data-driven fault diagnosis, [14] and [15] discuss power electronics applications. In particular, [15] uses the conditional entropy to interpret black-box controllers and remove abnormal training data. Finally, we highlight that there are few works which attempt to adapt XAI tools for dynamical systems, as shown in [16] using LIME.

In this paper, we aim at laying the first stone towards the development of *eXplainable data-driven control* (XDDC), leveraging the nature of physical and dynamical systems of interest in the field. Indeed, in control applications, it is likely to have both prior knowledge, e.g., from first principles, about the physical properties that we would like to recover in the data-driven model, and some prior information or constraint about dynamical properties of interest, like stability of both open-loop and closed-loop systems. Exploiting knowledge to improve explainability is not a completely new concept in the field of XAI, as shown by the *knowedge corpus* [17][1], and by Physics-Informed Neural Networks (PINNs) [18], which encode in the training process the prior knowledge of known differential equations as an additional regularization term. Despite this, the possibility of preserving some known property in the explainable models has not been

Both the authors are with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Via Ponzio 34/5, 20133 Milan, Italy. Email to: giorgio.riva@polimi.it.This paper is partially supported by FAIR (Future Artificial Intelligence Research) project, funded by the NextGenerationEU program within the PNRR-PE-AI scheme (M4C2, Investment 1.3, Line on Artificial Intelligence), by the Italian Ministry of Enterprises and Made in Italy in the framework of the project 4DDS (4D Drone Swarms) under grant no. F/310097/01-04/X56 and by the PRIN PNRR project P2022NB77E "A data-driven cooperative framework for the management of distributed energy and water resources" (CUP: D53D23016100001), funded by the NextGeneration EU program.

[1]Unstructured domain knowledge is used in the XAI system to match the semantics of ML models with human interpretable concepts, e.g., through textual explanation. This is not the usual situation in data-driven control applications, where systems are usually dynamical and knowledge can rarely be expressed via textual or visual information.

investigated from a control systems perspective yet. Thus, in this work, we define a novel concept of explainability tailored for data-driven control, adding prior knowledge in a systematic way but also setting a novel *property-preserving framework*. To strenghten the discussion, we address two case studies taken from the available literature, showing the effectiveness of the property-preserving paradigm to obtain explainable data-driven objects both in traditional system identification and model-based control and in more recent direct data-driven control system design.

The outline of the paper is as follows. in Section II, we introduce and discuss the property-preserving framework for XDDC. Section III presents the first case study, about data-driven control design for power split in hybrid battery packs for electric racing vehicles. In Section IV, the second case study is proposed, highlighting the role of preserving physical and stability properties in the identification of an electro-mechanical positioning system for control design. The paper is ended by some concluding remarks, pavying the way for future researches in this field.

II. Property-Preserving Explainable Data-Driven Control

In this section, we define the property-preserving framework for explainability in data-driven control. To this aim, we need to introduce a proper model taxonomy. Models ($\mathcal{M}$) are historically categorized as white-box, grey-box, and black-box, with different meanings with respect to ML. White-box models ($\mathcal{W}$) are fully built from first principles, from which controllers are designed using established model-based synthesis techniques. Grey-box models ($\mathcal{G}$) are located in between the model-based and data-driven world, where the structure of the model, e.g., the dynamical equations, is derived from first principles, while parameters from data. The same concept can be used for control design, where the parameters of suitable model-based controllers are identified from data, using experiments. Black-box models ($\mathcal{B}$), instead, are strictly related to ML models, where a distintion can be made between transparent ($\mathcal{T}$, e.g., linear parameter varying models) and opaque models ($\mathcal{O}$, e.g., echo-state neural networks). In this context, the explanations achieved via standard XAI methods might be either in conflict with the designer prior knowledge about the system, or incomplete. For this reason, we propose a further model categorization: the property-preserving model class ($\mathcal{P}$), whose members satisfy some specified physical and dynamical properties. Fig. 1 depicts the discussed model categorization in control, introducing the property-preserving class, which might intersect both transparent and opaque ones. Indeed, asymptotically stable linear time-invariant systems are examples of transparent property-preserving models, while echo-state networks, if equipped with enforced stability during random coefficients sampling, belong to opaque property-preserving ones. We highlight that, transparent property-preserving models represent the best possible scenario, merging a global, direct and specific explanation with property guarantees coming from the proposed paradigm, which defines *an additional explainability layer*.

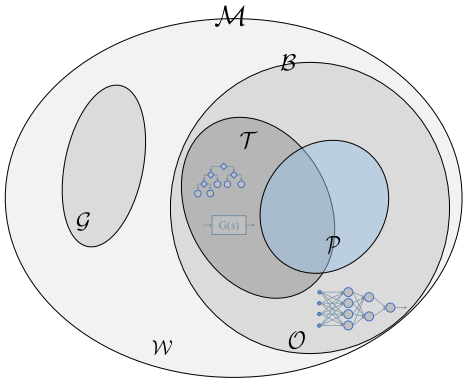

Fig. 1. Model taxonomy in data-driven control: $\mathcal{M}$ is the whole model class; $\mathcal{W}$, $\mathcal{G}$ and $\mathcal{B}$ include respectively white-box, grey-box and black-box models; and $\mathcal{T}$, $\mathcal{O}$, and $\mathcal{P}$ contain respectively transparent, opaque and property-preserving models.

Given the above intuition, we hereafter define the property-preserving framework for XDDC more formally. To this aim, we start from a rather general formulation of the model learning problem, which can be written as:

$$\min_{\mathrm{M}\in\mathcal{M}} \mathrm{E_{in}}\left(\mathrm{S},\mathrm{M}\right) + \lambda \cdot \mathrm{Compl}\left(\mathrm{M}\right). \tag{1}$$

In (1), $\mathrm{E_{in}}$ is the in-sample error between the true system S and the model $\mathrm{M} \in \mathcal{M}$, either white-box or black-box, while the last term is a regularization penalty, weighted by a user-defined coefficient λ, to handle over-fitting by limiting model complexity $\mathrm{Compl}\left(\mathrm{M}\right)$. In the standard XAI framework, explainability *ante-hoc* is achieved by restricting the search set to $\mathcal{T} \subset \mathcal{M}$, namely to the transparent models:

$$\min_{\mathrm{M}\in\mathcal{T}\subset\mathcal{M}} \mathrm{E_{in}}\left(\mathrm{S},\mathrm{M}\right) + \lambda \cdot \mathrm{Compl}\left(\mathrm{M}\right), \tag{2}$$

while post-hoc one is achieved downstreaming the training of the model. In data-driven control, the set of models of interest is defined by $\mathcal{B}$, the black-box ones. The original ML problem in (1) can be modified to include the property-preserving paradigm restricting the search set to $\mathcal{P} \subset \mathcal{B}$, as:

$$\min_{\mathrm{M}\in\mathcal{P}\subset\mathcal{B}} \mathrm{E_{in}}\left(\mathrm{S},\mathrm{M}\right) + \lambda \cdot \mathrm{Compl}\left(\mathrm{M}\right) \tag{3}$$

Eventually, this restriction can be achieved by means of suitable constraints from the original set $\mathcal{B}$, exploiting the prior knowledge on S:

$$\begin{aligned} &\min_{\mathrm{M}\in\mathcal{B}} \mathrm{E_{in}}\left(\mathrm{S},\mathrm{M}\right) + \lambda \cdot \mathrm{Compl}\left(\mathrm{M}\right) \\ &\quad \text{s.t.}: \\ &\quad \mathrm{M} \in \mathrm{prior}\left(\mathrm{S}\right) \end{aligned} \tag{4}$$

Addressing the learning problem outlined above from a broad theoretical perspective would be overly extensive and complex, especially at this early stage, given the diverse range of possible scenarios. Therefore, in this initial study, we will focus on demonstrating how specific dynamic properties can be maintained within a learning process and evaluating the

implications of framing the problem as proposed through two distinct and relevant case studies. A formal examination of this framework is currently underway as part of ongoing research and will be addressed in future publications.

III. Case Study 1: Controller Design for Power Split in Hybrid Battery Packs

In this first case study, we tackle the problem of learning an energy management strategy (EMS) for hybrid battery packs in racing electric vehicles, aiming to distribute power between available sources. This study builds upon [19] and [20], where [19] introduces an optimization framework for fully electric vehicles, and [20] extends it to hybrid battery packs. The design and control of hybrid battery packs are pertinent in automotive electrification trends, particularly in racing competitions driving technological advancements. In [20], an implicit EMS control logic is optimized for power split, necessitating the development of an explicit real-time implementable logic. Our study aims to learn this explicit policy from the implicit optimal one in a data-driven approach. We introduce the application and data generation process, discuss the use of regression trees for explainability, and compare results with neural networks using SHAP for post-hoc explanation.

As data generator mechanism, we leveraged the work in [20], where the optimal sizing of a hybrid battery pack for the Generation 3 (Gen 3) Formula E race car has been addressed. Starting from its detailed analysis, we selected the optimal hybrid battery layout with commercial cells, which is made up by the Kokam SLPB065 and the Saft VL5U, respectively for high energy and high power cells. Moreover, we stick with the same mission profile for sizing, i.e., the 2021 Rome ePrix, considering 23 laps as desired mileage. The optimal hybrid battery size resulted in 17 parallels for the high energy battery, and a single parallel for the high power one. The training dataset for our learning problem is made up by the battery power split obtained from such optimal battery size, which is shown in Fig 2. As additional dataset for testing

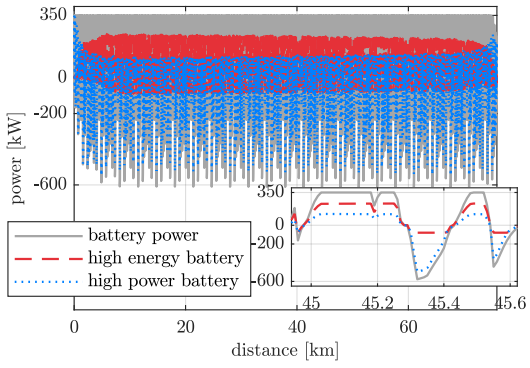

Fig. 2. Training dataset for the hybrid battery pack case study: input battery power request and optimal power split between the two batteries.

purposes, we computed the optimal power split with the same battery configuration for the 23 laps Valencia ePrix. We highlight that, in the following, the high energy battery is indicated by index 1, while the high power one with the index 2. Regarding the employed features, we selected all the available information that can be either measured or estimated by the actual control unit, namely: the desired battery power P_b, the State of Charge (SoC) of both batteries, i.e., SoC_1 and SoC_2, and the power limits of both batteries, respectively $P_{b,1}^{\text{low}}$, $P_{b,1}^{\text{high}}$, $P_{b,2}^{\text{low}}$, and $P_{b,2}^{\text{high}}$. Finally, given the constraint on the total request, only the power split on the high energy battery is learnt, obtaining equal power split errors, with opposite sign, between the two. Thus, the Root Mean Square Error (RMSE) computed based on the high energy battery, is representive of both.

Despite the main objective of the case study is to learn a policy approximating the implicit optimal solution, we also aim at achieving a logic which can be easily interpreted and understood by race engineers, for a reliable real-time implementation. For this reason, we selected linear regression trees [21] as model class, which guarantees a global, direct and specific explanation, as discussed in Section I. Indeed, the logic is characterized by binary decisions, each one based on a unique threshold on a single feature, and linear regressions for each leaf of the tree, which determine how the split is performed based on the input features. The training has been performed in Python 3.7 using the linear-tree[2] package, restricting the maximum depth to three and the minimum percentage of samples for each split at 2%, to improve explainability and mitigate overfitting. A pictorial representation for the obtained decision tree is reported in Fig. 3. As anticipated, with the chosen models we have

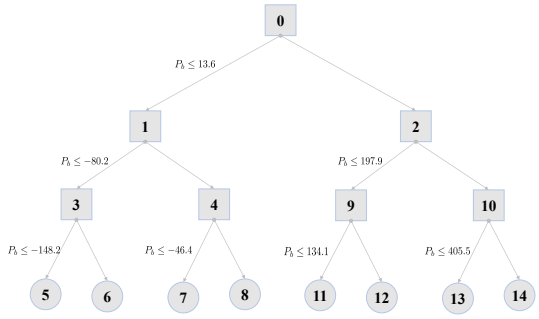

Fig. 3. Representation of the regression tree without property-preserving constraints.

a straightforward interpretation of the logic. However, the explanation obtained lacks reliability since binary decisions rely solely on the input battery power request value, without considering the state of the batteries. To enhance the explanation, we aim to incorporate prior knowledge about relevant physical phenomena using the property-preserving paradigm. As a first step, we introduce four additional binary variables $\left(B_1^{\text{sat,low}}, B_1^{\text{sat,high}}, B_2^{\text{sat,low}}, \text{and } B_2^{\text{sat,high}}\right)$, indicating whether the requested battery power exceeds the lower or upper power limits of each battery. Additionally, we

[2]https://pypi.org/project/linear-tree

enforce the dichotomy of traction and braking conditions, crucial for power split logic in race cars, by constraining the first binary decision based on the sign of the input battery power. This results in two separate sub-trees for braking and traction, as depicted in Fig. 4. In contrast to

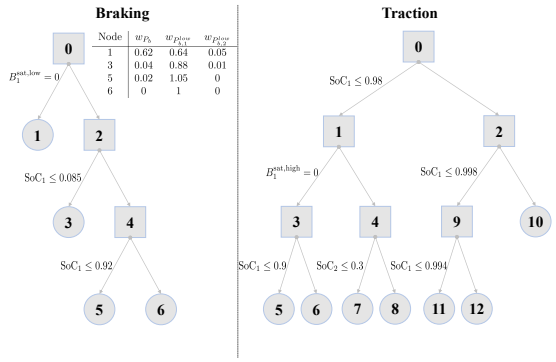

Fig. 4. Representation of the property-preserving regression tree: enforced split between traction and braking conditions. In braking, a table with the relevant regression weights for each leaf node is reported.

the previous model, decisions in both braking and traction conditions are now based on features related to the state of the batteries, resulting in more reliable outcomes. In the braking sub-tree, the first decision depends on whether the high-energy battery is saturated by the request, with subsequent leafs providing regression models approximated by $P_{b,1} = P_{b,1}^{\text{low}}$, as shown in the table of Fig. 4 reporting the relevant features' regression weights, indicating saturation to maximize regeneration capabilities. In the traction part, leafs 5 and 6 dominate, with the regression model approximated by $P_{b,1} = 0.6 \cdot P_b$, representing a simple power split between the two batteries. This illustrates how incorporating prior physical knowledge improves the reliability of explanations, even with established transparent XAI models.

For a comprehensive analysis, we compare our approach with feed-forward neural networks, serving as a benchmark for accuracy. To ensure a fair comparison, we select two different complexities: 1) a shallow neural network with 200 neurons (nn-full) for reference performance, and 2) a deep neural network with two layers, having 7 and 4 neurons respectively (nn-limited), to match the number of free parameters in our property-preserving regression tree. Both models are trained in Python 3.7 using the TensorFlow package, minimizing the mean square error via a stochastic gradient descent algorithm with learning rate 0.1. In Fig. 5, we compare the accuracy of all trained models, evaluated via RMSE, on both training and testing data. Overall, all models perform well, with relatively small power estimation errors, as depicted in Fig. 6 for the Valencia ePrix testing data. As expected, the more complex neural network (nn-full) performs best on the training data. Although the property-preserving regression tree, due to additional constraints, exhibits slightly lower performance compared to the standard one, it outperforms the neural network with equal complexity (nn-limited). In terms of explainability (as discussed in

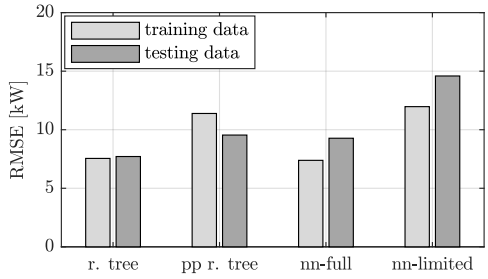

Fig. 5. Accuracy comparison on training and testing data. Compared models: 1) standard regression tree (r.tree), 2) property-preserving regression tree (pp r. tree), 3) complex neural network (nn-full), and 4) limited complexity neural network (nn-limited).

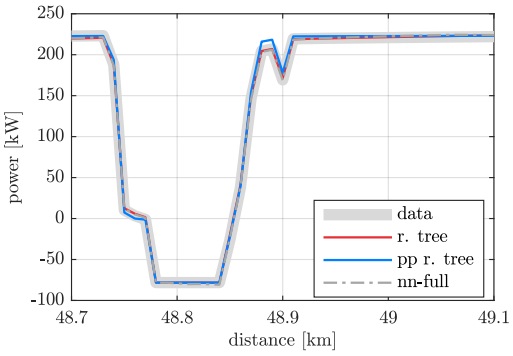

Fig. 6. Comparison of power split testing performance. Compared models: 1) standard regression tree (r.tree), 2) property-preserving regression tree (pp r. tree), 3) complex neural network (nn-full).

Section I), opaque models' logic can only be interpreted post-hoc. For a direct comparison with the explanations obtained for regression trees, we utilize SHAP [8], a local method based on specific predictions. Results are nearly identical for both opaque models, and here we present those for nn-full. In Fig. 7, shapley values for each input feature for one braking (left) and one traction (right) prediction are displayed, revealing the significant influence of the input battery power request. The distribution of shapley values for all predictions is depicted in Fig. 8, confirming the pivotal role of this input feature. These observations align with those of the standard regression tree, where all binary decisions were based on the input power request. In conclusion, these results demonstrate how the property-preserving paradigm enables a more meaningful interpretation of the control logic, albeit with a slight accuracy loss in this case study.

IV. Case Study 2: Model-based control of an Electro-Mechanical Positioning System

In this second case study, we transition from direct data-driven control design to a classical model-based control design known as "indirect data-driven design." Using a set of nonlinear benchmarks, we tackle the Electro-Mechanical Positioning System (EMPS) problem to identify a dynamical

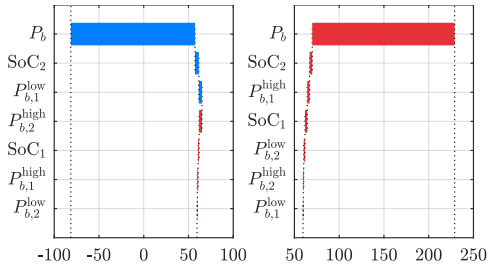

Fig. 7. Shapley values of the benchmark neural network (nn-full) in two specific predictions: one braking (left) and one traction (right) condition.

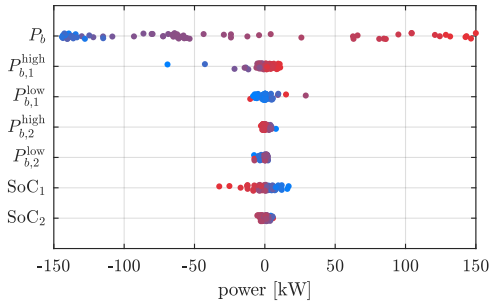

Fig. 8. Distribution of shapley values of the benchmark neural network (nn-full) on the whole testing dataset.

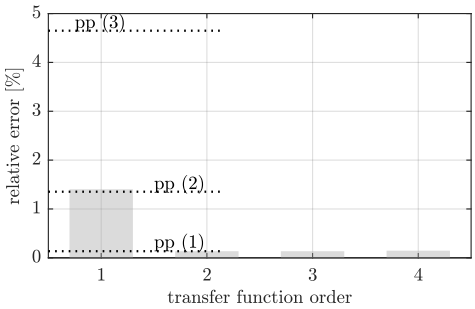

Fig. 9. Relative position estimation error as a function of the transfer function model order.

model suitable for control design, as presented in [22]. Two identification datasets, for training and testing, are provided along with a grey-box estimation approach based on the inverse dynamic identification model (IDIM). In the context of data-driven control, we utilize both transparent and opaque black-box models. For transparent models, a linear time-invariant model in transfer function representation is selected, offering explainability akin to the benchmark grey-box model. As representatives of opaque models, a shallow feedforward neural network with 5 neurons is employed to learn the mapping from past positions and input values to the current actuator position. The optimal number of neurons and past values is determined through sensitivity analysis on the training data. To ensure a fair comparison, all models are trained to minimize simulation error over the entire training data horizon using the MATLAB function *fmincon*. To show the impact of the property-preserving paradigm, the class of transparent models has been endowed with three incremental levels of prior knowledge about the system: 1) we impose a second-order transfer function, knowing the mechanical nature of the system, 2) we enforce the presence of a pure integrator in the system, from speed to position, due to the absence of a load [22], and 3) we compel the remaining eigenvalue to be asymptotically stable, to preserve open-loop stability. As first step, we optimized the order of the transfer function model when no prior knowledge is used. Fig. 9 shows the obtained accuracy on the training data, where performance are expressed in terms of relative position error, as in [22], and compared with that of the three incremental property-preserving models (pp). It is visible that the best order for the transfer function, as trade-off between accuracy and complexity, is two, which represents exactly the first level of property-preserving. Thus, from now on, we focus only on the three levels of property-preserving models, whose general transfer function expression is $G(s) = \frac{b_1 s + b_0}{s^2 + a_1 s + a_0}$. The optimal values of the free parameters of the three models are reported in Tab I. We can highlight the impact of the preserved properties: levels two and three have zero value a_0 coefficient to guarantee the presence of the integrator, and the third one shows a positive coefficient a_1, which corresponds to a negative eigenvalue. The performance of the different models are compared in

TABLE I

OPTIMIZED TRANSFER FUNCTION PARAMETERS FOR THE THREE PROPERTY-PRESERVING LEVELS.

	b_0	b_1	a_0	a_1
tf pp (1)	0.39	-3.48×10^{-4}	-0.28	4.70
tf pp (2)	8.62×10^{-4}	7.68×10^{-2}	0	-5.57×10^{-2}
tf pp (3)	0.01	6.58×10^{-2}	0	2.08×10^{-7}

Fig. 10 on the testing data. The neural network results to be the most accurate model, showing comparable performance with the first level of property-preserving transfer functions, i.e., of second order. Moreover, the second level of property-preserving models outperforms the grey-box benchmark in [22], which in turn defeats the third level one. Accuracy on the testing data is also in shown in Fig. 11, where the different models are compared in time-domain. Regarding explainability, the situation is reversed, and the latter two models show the highest level of physical interpretation. In addition, despite the limited accuracy loss, the third property-preserving model attains another important advantage with respect to all the other black-box models: indeed, preserved open-loop stability allows the usage of more standard control synthesis techniques, like bode criterion, for the design of a closed-loop position controller.

V. CONCLUSIONS

In this paper, we propose a novel framework for achieving explainable data-driven control by integrating established principles into the control process. This involves imposing constraints on either the control-oriented model or the direct derivation of feedback actions from data, as seen in direct

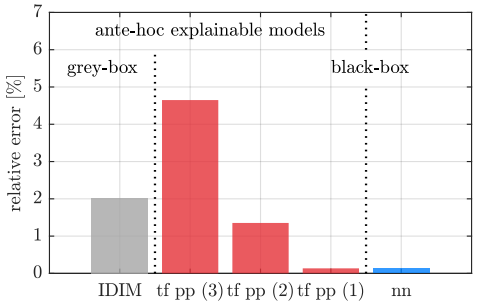

Fig. 10. Accuracy comparison on the testing data. Compared models: 1) benchmark model-based approach (IDIM), 2) third level property-preserving transfer function, 3) second level property-preserving transfer function, 4) first level property-preserving transfer function, and (5) benchmark neural network.

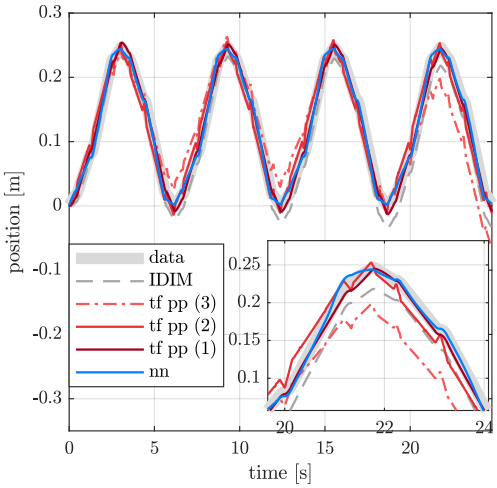

Fig. 11. Time-domain position estimation comparison on testing data. Compared models: 1) benchmark model-based approach (IDIM), 2) third level property-preserving transfer function, 3) second level property-preserving transfer function, 4) first level property-preserving transfer function, and (5) benchmark neural network.

data-driven control methods. Despite resulting in black-box laws, our approach offers transparency for easy interpretation. Due to the early stage of our development, a detailed theoretical discussion is considered overly intricate. Instead, we present a preliminary exploration demonstrating the application of this approach in both direct and indirect data-driven design scenarios. Using experimental data, we address two modern control challenges to showcase the methodology's applicability. This work establishes the groundwork for further examination of feedback loop explainability to enhance transparency in complex control systems.

Future efforts will focus on automatically embedding formal properties, extensive experimental testing, and theoretical analysis of the resultant loop.

References

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

[2] J. W. Goodell, S. Kumar, W. M. Lim, and D. Pattnaik, "Artificial intelligence and machine learning in finance: Identifying foundations, themes, and research clusters from bibliometric analysis," *Journal of Behavioral and Experimental Finance*, vol. 32, p. 100577, 2021.

[3] Y. Ma, Z. Wang, H. Yang, and L. Yang, "Artificial intelligence applications in the development of autonomous vehicles: A survey," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 2, pp. 315–329, 2020.

[4] W. Tong, A. Hussain, W. X. Bo, and S. Maharjan, "Artificial intelligence for vehicle-to-everything: A survey," *IEEE Access*, vol. 7, pp. 10 823–10 843, 2019.

[5] W. J. Murdoch, C. Singh, K. Kumbier, R. Abbasi-Asl, and B. Yu, "Interpretable machine learning: definitions, methods, and applications," *arXiv preprint arXiv:1901.04592*, 2019.

[6] N. Burkart and M. F. Huber, "A survey on the explainability of supervised machine learning," *Journal of Artificial Intelligence Research*, vol. 70, pp. 245–317, 2021.

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

[9] 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.

[10] M. Quade, T. Isele, and M. Abel, "Machine learning control—explainable and analyzable methods," *Physica D: Nonlinear Phenomena*, vol. 412, p. 132582, 2020.

[11] P. Ashok, M. Jackermeier, P. Jagtap, J. Křetínskỳ, M. Weininger, and M. Zamani, "dtcontrol: Decision tree learning algorithms for controller representation," in *Proceedings of the 23rd international conference on hybrid systems: Computation and control*, 2020, pp. 1–7.

[12] P. Ashok, M. Jackermeier, J. Křetínskỳ, C. Weinhuber, M. Weininger, and M. Yadav, "dtcontrol 2.0: Explainable strategy representation via decision tree learning steered by experts," in *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*. Springer, 2021, pp. 326–345.

[13] D. Šoberl and I. Bratko, "Learning explainable control strategies demonstrated on the pole-and-cart system," in *Advances and Trends in Artificial Intelligence. From Theory to Practice: 32nd International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2019, Graz, Austria, July 9–11, 2019, Proceedings 32*. Springer, 2019, pp. 483–494.

[14] A. Beattie, P. Mulink, S. Sahoo, I. T. Christou, C. Kalalas, D. Gutierrez-Rojas, and P. H. Nardelli, "A robust and explainable data-driven anomaly detection approach for power electronics," in *2022 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*. IEEE, 2022, pp. 296–301.

[15] S. Sahoo, H. Wang, and F. Blaabjerg, "On the explainability of black box data-driven controllers for power electronic converters," in *2021 IEEE Energy Conversion Congress and Exposition (ECCE)*. IEEE, 2021, pp. 1366–1372.

[16] 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.

[17] X.-H. Li, C. C. Cao, Y. Shi, W. Bai, H. Gao, L. Qiu, C. Wang, Y. Gao, S. Zhang, X. Xue *et al.*, "A survey of data-driven and knowledge-aware explainable ai," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 1, pp. 29–49, 2020.

[18] S. Cuomo, V. S. Di Cola, F. Giampaolo, G. Rozza, M. Raissi, and F. Piccialli, "Scientific machine learning through physics–informed neural networks: Where we are and what's next," *Journal of Scientific Computing*, vol. 92, no. 3, p. 88, 2022.

[19] G. Riva, S. Radrizzani, G. Panzani, M. Corno, and S. M. Savaresi, "An optimal battery sizing co-design approach for electric racing cars," *IEEE Control Systems Letters*, vol. 6, pp. 3074–3079, 2022.

[20] S. Radrizzani, G. Riva, G. Panzani, M. Corno, and S. M. Savaresi, "Optimal sizing and analysis of hybrid battery packs for electric racing cars," *IEEE Transactions on Transportation Electrification*, 2023.

[21] W.-Y. Loh, "Classification and regression trees," *Wiley interdisciplinary reviews: data mining and knowledge discovery*, vol. 1, no. 1, pp. 14–23, 2011.

[22] A. Janot, M. Gautier, and M. Brunot, "Data set and reference models of emps," in *Nonlinear System Identification Benchmarks*, 2019.