POLITECNICO DI TORINO
Repository ISTITUZIONALE

A kernel-based approach to physics-informed nonlinear system identification

Original
A kernel-based approach to physics-informed nonlinear system identification / Donati, C., Mammarella, M., Calafiore, G.C., Dabbene, F., Lagoa, C., Novara, C.. - In: IEEE TRANSACTIONS ON AUTOMATIC CONTROL. - ISSN 0018-9286. - ELETTRONICO. - 71:7(2026), pp. 4921-4928. [10.1109/tac.2026.3663111]

Availability:
This version is available at: 11583/3008460 since: 2026-03-10T00:00:42Z

Publisher:
IEEE

Published
DOI:10.1109/tac.2026.3663111

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

14 September 2026

A Kernel-Based Approach to Physics-Informed Nonlinear System Identification

Cesare Donati, *Graduate Student Member, IEEE*, Martina Mammarella, *Senior Member, IEEE*, Giuseppe C. Calafiore, *Fellow, IEEE*, Fabrizio Dabbene, *Fellow, IEEE*, Constantino Lagoa, *Member, IEEE*, and Carlo Novara, *Senior Member, IEEE*

***Abstract*—This article presents a kernel-based framework for physics-informed nonlinear system identification that extends kernel-based techniques, accounting for unmodeled dynamics, to seamlessly embed partially known physics-based models, improving parameter estimation and overall model accuracy. The two models' components are identified from data simultaneously, thereby minimizing a suitable cost that balances the relative importance of the physical and the black-box parts of the model. In addition, nonlinear state smoothing is employed to address scenarios involving state-space models with not-fully measurable states. Numerical simulations on an experimental benchmark system demonstrate the effectiveness of the proposed approach, achieving up to 51% reduction in simulation root mean square error compared to physics-only models and 31% performance improvement over state-of-the-art identification techniques.**

***Index Terms*—Estimation, grey-box modeling, kernel methods, nonlinear systems identification.**

I. Introduction

Nonlinear system identification plays a crucial role in engineering, aiming to construct models that accurately represent the complex dynamics of physical systems using measured data. Traditional approaches often rely on parametric models derived from physical principles (also referred to as off-white models) [1] or more flexible representations (black-box models), such as neural networks [2] or nonlinear basis function combinations [3]. Recently, the integration of these two approaches has been proposed to tackle the challenge of compensating for unmodeled dynamics in physical models that arise in real-world applications (see, e.g., [4]). Within this context, some identification approaches follow a two-step process: first, they estimate the physical parameters while assuming no unmodeled dynamics, and then they introduce corrections to model the resulting *discrepancy* [5], [6]. This strategy inevitably produces biased physical parameter estimates, which need to be handled a posteriori by compensating them via the black-box component of the model. Therefore, the latter term must not only account for modeling errors but also compensate for the bias error induced by the parametric model identification phase.

An alternative perspective is presented in, e.g., [7] and [8], by modeling unmodeled dynamics explicitly from the beginning and estimating both the physical parameters and correction terms simultaneously. In this way, the interference between the two is minimized, leading to a more accurate and reliable identification process that prevents biased parameter estimates. In this context, sparsification is essential to ensure that corrections remain interpretable and do not overshadow the underlying physical model. However, despite their effectiveness in identifying nonlinear models, a key limitation of these methods is the need for a careful selection of appropriate basis functions that can adequately capture the underlying unmodeled system dynamics [9].

Kernel methods are a class of nonparametric machine learning techniques, able to provide a powerful framework for overcoming these challenges by enabling the construction of regularized models directly from data, without the need for explicitly defining basis functions [10]. These methods are widely used in the context of input–output system identification (see, e.g., [11], [12]), where the relationship between inputs and outputs is learned directly from measured data. However, the conventional kernel approaches do not incorporate physical system knowledge, which on the other hand can be crucial for developing interpretable and reliable models, featuring also improved generalization capabilities. In this work, we aim to fill this gap by proposing a novel identification framework that embeds kernel methods with available physics-based models. The approach leverages kernel-based function approximations to systematically compensate for unmodeled dynamics, while preserving the interpretability of the physical part of the model. Unlike traditional methods that rely on predefined (and often heuristically chosen) basis function dictionaries, the proposed formulation adapts directly to the data. Thus, by leveraging the representer theorem [13], the proposed framework provides a regularized, data-driven functional approximation mechanism that eliminates the need for manual selection of the basis functions while preserving both interpretability and accuracy through the embedded physical structure.

Moreover, we rely on a more general state-space setting to encompass a broader class of dynamical systems by extending the proposed method beyond traditional input-output identification models (e.g., NARX) [14]. In fact, many physical systems are better described by state-space models, which explicitly capture system dynamics over time [9]. This formulation also unifies various prediction models, including nonlinear output error, ARMAX, and ARX models, and serves as a foundation for many controller and observer design methods, making it especially valuable in system identification. Unlike static regression models, a state-space formulation accounts for the evolution of hidden states, requiring estimation of both system parameters and unmeasured state trajectories. One common approach to this challenge is to rely on multistep identification methods, where

Received 20 April 2025; revised 20 October 2025; accepted 2 February 2026. Date of publication 9 February 2026; date of current version 1 July 2026. This work was supported in part by the project "Space It Up – Spoke 1", funded by the Italian Space Agency (ASI) and in part by the Ministry of University and Research (MUR), under the Italian National Recovery and Resilience Plan (PNRR). Recommended by Associate Editor L. Bako. *(Corresponding author: Cesare Donati.)*

Cesare Donati is with the Department of Electronics and Telecommunications, Politecnico di Torino, 10129 Torino, Italy, and also with CNR-IEIIT, 10129 Torino, Italy (e-mail: cesare.donati@polito.it).

Martina Mammarella and Fabrizio Dabbene are with CNR-IEIIT, 10129 Torino, Italy.

Giuseppe C. Calafiore and Carlo Novara are with the Department of Electronics and Telecommunications, Politecnico di Torino, 10129 Torino, Italy.

Constantino Lagoa is with EECS, Pennsylvania State University, University Park, PA 16802 USA.

Digital Object Identifier 10.1109/TAC.2026.3663111

© 2026 The Authors. This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see http://creativecommons.org/licenses/by/4.0/

unmeasured states are recursively estimated through repeated model evaluation (see, e.g., [15]). While this strategy naturally estimates latent states by iterating the estimation model, it often makes the optimization challenging due to its strong nonlinear parameter dependencies. Moreover, extending the kernel-based framework to multistep settings introduces further complexity. To circumvent these issues, we adopt an alternative strategy inspired by [16], in which prior state estimates, derived from available data, are used within prediction-based state-space optimization problems. To this end, we combine an unscented Kalman filter (UKF) [17] with an unscented Rauch–Tung–Striebel smoother (URTSS) [18] to reconstruct the hidden state trajectories, enabling kernel-based model embedding in a state-space setting. Last, the effectiveness of the proposed approach is validated through an academic example and an experimental benchmark, showcasing its advantages over state-of-the-art techniques in both predictive accuracy and simulation performance.

The rest of this article is organized as follows. Section II introduces the nonlinear function estimation using kernel theory. The main contribution of this article, i.e., embedding the parameterized physical models with data-driven kernels to account for unmodeled dynamics, is presented in Section III. Then, Section IV extends the kernel-based framework to state-space systems via nonlinear state smoothing. The numerical results are discussed in Section V. Finally, Section VI concludes this article.

II. Kernel-Based Approximation

In this section, we provide a brief introduction to nonlinear function approximation using kernels, which represents the core foundation for the main results presented in this article. First, we introduce two definitions related to kernels.

Definition 1 (Positive definite kernel [13]): Let $\mathcal{X}$ be a nonempty set. A real-valued, continuous, symmetric function $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a positive-definite kernel (on $\mathcal{X}$) if $\sum_{i=1}^{n}\sum_{j=1}^{n} c_i c_j \kappa(x_i, x_j) \geq 0$ holds $\forall\, n \in \mathbb{N}, x_1, \ldots, x_n \in \mathcal{X}, c_1, \ldots, c_n \in \mathbb{R}$.

Definition 2 (Reproducing kernel Hilbert space [19]): Let $\mathcal{H}$ be a Hilbert space of real-valued functions defined on a nonempty set $\mathcal{X}$, with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$. A function $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a reproducing kernel of $\mathcal{H}$, and $\mathcal{H}$ is a reproducing kernel Hilbert space (RKHS) on $\mathcal{X}$ if the following conditions hold: (i) $\forall\, x \in \mathcal{X}$, $\kappa(\cdot, x) \in \mathcal{H}$; and (ii) the *reproducing property* holds, i.e., $\forall\, x \in \mathcal{X}$, $h \in \mathcal{H}$, $\langle h(\,\cdot\,), \kappa(\cdot, x)\rangle_{\mathcal{H}} = h(x)$.

From the reproducing property in Definition 2.(ii), we observe that the value of h in x can be represented as an inner product in the feature space. Hence, applying this property to the kernel κ, for any $x, x' \in \mathcal{X}$, we have that $\kappa(x, x') = \langle \kappa(\cdot, x), \kappa(\cdot, x')\rangle_{\mathcal{H}}$. Moreover, according to the Moore–Aronszajn theorem [19], every positive-definite kernel κ uniquely defines an RKHS $\mathcal{H}$ in which κ serves as the reproducing kernel. Conversely, each RKHS is associated with a unique positive-definite kernel. Given a kernel κ, we now consider a nonlinear input–output relation given by an unknown nonlinear function $g : \mathcal{X} \to \mathbb{R}$

$$y = g(x) + e \tag{1}$$

where $x \in \mathcal{X}, y \in \mathbb{R}$ are the input and output, respectively, g is assumed to belong to the native RKHS $\mathcal{H}$ associated with the given kernel κ, and $e \in \mathbb{R}$ is an error term which represents measurement noise, as well as possible structural errors on g. Let $\mathcal{D} = \{(x_1, y_1), \ldots, (x_T, y_T)\}$ be a sequence of given T input–output data, collected from system (1). The goal is to find an estimate $\hat{g}$ of the function g, accurately representing the observed data while ensuring that, for any new pair of data (x, y), the predicted value $\hat{g}(x)$ remains close to y.

A standard approach to estimate g using the dataset $\mathcal{D}$ involves minimizing a loss function that combines a quadratic data-fit term (i.e., the prediction error) with a regularization one. Hence, the function g can be estimated by solving the well-known kernel ridge regression (KRR) problem [20]

$$\hat{g} = \arg\min_{g \in \mathcal{H}} \sum_{t=1}^{T} (y_t - g(x_t))^2 + \gamma \|g\|_{\mathcal{H}}^2 \tag{2}$$

where $\gamma \in \mathbb{R}$ is a tradeoff weight that balances data-fit and regularization, and $\|g\|_{\mathcal{H}} = \sqrt{\langle g, g\rangle_{\mathcal{H}}}$ is the norm in $\mathcal{H}$, introduced by the inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$. Then, the *representer theorem* [13] guarantees the uniqueness of the solution to (2), expressed as a sum of T basis functions determined by the kernel, each function weighted by coefficients obtained through the solution of a system of linear equations (see, e.g., [21]). In particular, given a positive-definite, real-valued kernel $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, and defining the kernel matrix $\mathbf{K} \in \mathbb{R}^{T,T}$ with the (i, j) entry defined as $\mathbf{K}_{ij} \doteq \kappa(x_i, x_j)$, and $Y \doteq [y_1, \ldots, y_T]^\top$, the application of the representer theorem yields $\hat{g}$ in closed form as follows:

$$\hat{g}(x) = \sum_{j=1}^{T} \omega_j \kappa(x, x_j) \quad \forall x. \tag{3}$$

Thus, considering (3) and solving KRR problem (2), the vector of weights $\omega = [\omega_1, \ldots, \omega_j, \ldots, \omega_T]^\top$ is defined as $\omega = (\mathbf{K} + \gamma \mathbf{I}_T)^{-1} Y$, with $\mathbf{I}_T$ the identity matrix of size T.

The result of this theorem is relevant as it demonstrates that a broad class of learning problems admits solutions that can be expressed as expansions of the training data. Building on this result, the next section explores how the representer theorem extends to the problem of embedding parameterized physical models with data-driven kernels to account for unmodeled dynamics and to estimate interpretable parameters.

III. Kernel-Based Model Embedding

Let us now consider a nonlinear map of the form

$$y = f(x, \bar{\theta}) + \Delta(x) + e \tag{4}$$

where $f : \mathcal{X} \times \Theta \to \mathbb{R}$ is a known function derived, e.g., from physical principles, and parametrized in $\bar{\theta} \in \Theta \subseteq \mathbb{R}^{n_\theta}$, whereas $\Delta : \mathcal{X} \to \mathbb{R}$ is an unknown term representing, e.g., modeling errors, uncertainties, or dynamic perturbations. Given a set of T input–output data $\mathcal{D} = \{(x_1, y_1), \ldots, (x_T, y_T)\}$, collected from a realization of (4) with true parameters $\bar{\theta} \in \Theta$, the goal is to find an estimate $\theta^\star$ of $\bar{\theta}$ and a black-box approximation $\delta : \mathcal{X} \to \mathbb{R}$ of $\Delta(x)$ by solving the following optimization problem:

$$(\theta^\star, \delta^\star) = \arg\min_{\theta \in \Theta, \delta \in \mathcal{H}} \sum_{t=1}^{T} \left[y_t - \hat{y}_t(\theta, \delta)\right]^2 + \gamma \|\delta\|_{\mathcal{H}}^2 \tag{5}$$

where $\hat{y}_t = f(x_t, \theta) + \delta(x_t)$ is the prediction of y_t at time t, and the notation $\hat{y}_t \equiv \hat{y}_t(\theta, \delta)$ is used to stress its dependencies to θ and δ. Note that this formulation is physics-informed in the sense that the parametric component $f(x_t, \theta)$ embeds known physical relations, while the nonparametric kernel term $\delta(x)$ accounts only for unmodeled effects or discrepancies with respect to the physical prior. This contrasts with purely data-driven kernel regression (2) approaches, where no structural knowledge is incorporated.

Remark 1 (On the role of γ): The hyperparameter γ in (5) plays a central role in the proposed framework and, as in (2), it acts as a classical regularization weight, balancing data fit and model complexity. It can also be interpreted as a tradeoff parameter between the physics-based model $f(x, \theta)$ and the nonparametric correction $\delta(x)$. This balance,

inherent to any regularization-based method (see, e.g., [22]), requires a proper tuning of γ. For instance, it can be effectively handled through specifically designed selection procedures, such as k-fold cross-validation or validation-based tuning, as adopted in this work. In practice, larger values of γ may enforce stronger adherence to the physical model, promoting smaller and smoother corrections, but possibly suppressing the capability of $\delta(x)$ to represent the residuals. Conversely, smaller values of γ allow to capture finer effects by increasing the influence of $\delta(x)$, which, however, may potentially dominate over the physical part of the model.

The optimization problem in (5) aims to estimate the vector of physical parameters associated with the known component of the model $f(x,\theta)$ while simultaneously identifying a function $\delta(x)$ that captures the unmodeled term $\Delta(x)$. This approach embeds available prior physical knowledge $f(x,\theta)$, allows the identification of interpretable parameters $\bar{\theta}$, and systematically compensates for unmodeled effects $\Delta(x)$, ensuring a more comprehensive and structured representation of the system. Assuming that the unknown term $\Delta(x)$ belongs to the RKHS $\mathcal{H}$ associated with the chosen kernel indicates that the solution $\delta^\star$ will admit a kernel representation. Moreover, it also implies that $\Delta(x)$ can be effectively approximated using a finite number of kernel evaluations parametrized by the observed data points, as stated in Definition 2. This assumption is common in nonparametric regression [21] and provides a well-posed framework for learning unmodeled dynamics while ensuring regularization and generalization properties. Moreover, although restricting $\Delta(x)$ to a reproducing space, the RKHSs are flexible enough to approximate a broad class of nonlinear functions [23], making this assumption reasonable in many practical systems identification scenarios.

The primary goal of the identification process is to accurately estimate the physical parameters $\bar{\theta}$ entering the physical model $f(x,\theta)$. The kernel-based representation of $\Delta(x)$ captures and compensates for unmodeled dynamics while preserving the underlying physical structure. This approach ensures that the learned correction term complements the physics-based model rather than overshadowing it. The following key result extends the representer theorem to the system identification framework under consideration.

Theorem 1 (Kernel-based model embedding): Suppose that a nonempty set $\mathcal{X}$, a positive-definite real-valued kernel κ on $\mathcal{X}\times\mathcal{X}$, a dataset $\mathcal{D}=\{(x_1,y_1),\dots,(x_T,y_T)\}\in\mathcal{X}\times\mathbb{R}$, and a function $f:\mathcal{X}\times\Theta\to\mathbb{R}$, parametrized in $\theta\in\Theta\subseteq\mathbb{R}^{n_\theta}$ are given. Let us introduce the following functions:

$$\Gamma(\theta)\doteq[f(x_1,\theta),\dots,f(x_T,\theta)]^\top \tag{6a}$$

$$\omega(\theta)=(\mathbf{K}+\gamma\mathbf{I}_T)^{-1}(Y-\Gamma(\theta)) \tag{6b}$$

where $\mathbf{K}$ is the kernel matrix associated to κ and $\mathcal{D}$, having $\mathbf{K}_{ij}=\kappa(x_i,x_j)$. Then, Problem (5) admits a solution $(\theta^\star,\ \delta^\star)$ of the form

$$\theta^\star=\arg\min_{\theta\in\Theta}\sum_{t=1}^{T}\Big(y_t-f(x_t,\theta)-\mathbf{K}_t^\top\omega(\theta)\Big)^2+\gamma\omega(\theta)^\top\mathbf{K}\omega(\theta) \tag{7a}$$

$$\delta^\star(x)=\sum_{j=1}^{T}\omega_j^\star\kappa(x,x_j),\ \omega_j^\star\doteq\omega_j(\theta^\star) \tag{7b}$$

with $\mathbf{K}_t^\top$ the tth row of $\mathbf{K}$.

Proof: Given (5) and $\hat{y}_t=f(x_t,\theta)+\delta(x_t)$, define $J(\theta,\delta)=\sum_{t=1}^{T}[y_t-f(x_t,\theta)-\delta(x_t)]^2+\gamma\|\delta\|_{\mathcal{H}}$, such that (5) is $(\theta^\star,\delta^\star)=\arg\min_{\theta\in\Theta,\delta\in\mathcal{H}}J(\theta,\delta)$. Considering that, $\min_{\theta\in\Theta,\delta\in\mathcal{H}}J(\theta,\delta)=\min_{\theta\in\Theta}\min_{\delta\in\mathcal{H}}J(\theta,\delta)$, we have that a minimizer to (5) must satisfy

$$\delta^\star(\cdot)=\left(\arg\min_{\delta\in\mathcal{H}}J(\theta,\delta)\right)|_{\theta=\theta^\star} \tag{8a}$$

$$\theta^\star=\arg\min_{\theta\in\Theta}p(\theta) \tag{8b}$$

with $p(\theta)\doteq\min_{\delta\in\mathcal{H}}J(\theta,\delta)$. Here, the inner minimization problem is solved with respect to δ and represents a standard KRR problem. Indeed, considering $\tilde{y}_t\doteq y_t-f(x_t,\theta)$, we have

$$\delta^\star(\cdot,\theta)=\arg\min_{\delta\in\mathcal{H}}\sum_{t=1}^{T}[\tilde{y}_t-\delta(x_t)]^2+\gamma\|\delta\|_{\mathcal{H}}^2. \tag{9}$$

By the representer theorem [13], the optimal solution to (9) is

$$\delta^\star(x,\theta)=\sum_{j=1}^{T}\omega_j(\theta)\kappa(x,x_j). \tag{10}$$

Considering (10) and solving (9) as in [20] yields the weight vector $\omega(\theta)$ as $\omega=(\mathbf{K}+\gamma\mathbf{I}_T)^{-1}\tilde{Y}$, being $\tilde{Y}\doteq[\tilde{y}_1,\dots,\tilde{y}_T]^\top$. Thus, (6b) is obtained given (6a) and substituting $\tilde{y}_t\doteq y_t-f(x_t,\theta)$ in $\tilde{Y}$. Moreover, considering the function $p(\theta)$ in (8b), we have $p(\theta)=\min_{\delta\in\mathcal{H}}J(\theta,\delta)=J(\theta,\delta^\star(\cdot,\theta))$, which, substituting (10), simplifies to

$$p(\theta)=\sum_{t=1}^{T}(y_t-f(x_t,\theta)-\mathbf{K}_t^\top\omega(\theta))^2+\gamma\omega(\theta)^\top\mathbf{K}\omega(\theta) \tag{11}$$

noting that

$$\begin{aligned}\|\delta\|_{\mathcal{H}}^2&=\left\langle\sum_{i=1}^{T}\omega_i(\theta)\kappa(x,x_i),\sum_{j=1}^{T}\omega_j(\theta)\kappa(x,x_j)\right\rangle_{\mathcal{H}}\\&=\sum_{i=1}^{T}\sum_{j=1}^{T}\omega_i(\theta)\omega_j(\theta)\langle\kappa(x,x_i),\kappa(x,x_j)\rangle_{\mathcal{H}}\\&=\sum_{i=1}^{T}\sum_{j=1}^{T}\omega_i(\theta)\omega_j(\theta)\kappa(x_i,x_j)=\omega(\theta)^\top\mathbf{K}\omega(\theta)\end{aligned}$$

from linearity of the inner product and the reproducing property in Definition 2.b. Thus, we obtain (7) by substituting the solution to (8b), with $p(\theta)$ given by (11), into (10), which concludes the proof. □

Theorem 1 establishes that the optimal solution to estimation problem (5) can be formulated using kernel-based functions. In particular, the unmodeled component $\Delta(x)$ is approximated by $\delta(x)$, defined as a linear combination of kernel evaluations parametrized by the observed data points, thus leading to the following predictive model:

$$\hat{y}=f(x,\theta^\star)+\delta^\star(x)=f(x,\theta^\star)+\sum_{j=1}^{T}\omega_j^\star\kappa(x,x_j)$$

representing the optimal solution to Problem (5). Theorem 1 is particularly relevant as, in contrast to standard KRR, it enables the explicit incorporation of prior physical knowledge alongside the adaptability of kernel methods, avoiding the use of heuristically chosen basis functions.

Remark 2 (On hyperparameter tuning): Clearly, kernel methods still involve hyperparameters (e.g., the kernel bandwidth σ in Gaussian and Laplacian kernels), which are typically tuned heuristically or via validation. This issue, however, is not unique to kernels: dictionary-based methods also require hyperparameter choices, such as the regularization weights that promote sparsity, as well as parameters embedded in the basis functions themselves [8], concluding that some level of hyperparameter tuning is unavoidable in both approaches. Nevertheless, kernel-based models generally rely on a smaller number of hyperparameters, which simplifies the identification procedure compared to dictionary-based alternatives.

A. Affine-in-Parameters Models

A relevant special case arises when the physics-based function $f(x,\theta)$ is affine in θ. In this setting, the map (4) becomes

$$y = f_0(x) + f(x)^\top \bar{\theta} + \Delta(x) + e \tag{12}$$

where $f_0 : \mathcal{X} \to \mathbb{R}$, $f : \mathcal{X} \to \mathbb{R}^{n_\theta}$, and $\bar{\theta} \in \Theta \in \mathbb{R}^{n_\theta}$. Thus, optimization problem (7a) simplifies significantly, leading to a convex optimization problem with a closed-form solution, as formalized in the following theorem.

Theorem 2 (Closed-form solution of (7)): Consider the same setup of Theorem 1. Define $F(x) \doteq [f(x_1),\dots,f(x_T)]^\top \in \mathbb{R}^{T,n_\theta}$ and $Y_0 \doteq [y_1 - f_0(x_1),\dots,y_T - f_0(x_T)]^\top \in \mathbb{R}^T$. Assume $F(x)$ is full column rank. If the system model in (4) is affine in θ, as in (12), then the solution of (7a) is given by

$$\theta^\star = \left(F(x)^\top \Psi F(x)\right)^{-1} F(x)^\top \Psi Y_0 \tag{13}$$

with

$$\Psi \doteq (\mathbf{K} + \gamma \mathbf{I}_T)^{-1} \tag{14}$$

where $\mathbf{K}$ is the kernel matrix associated to κ and $\mathcal{D}$, and γ is the weight controlling the regularization of $\delta(\cdot)$ (7b).

Proof: Considering (7a) applied to (12), we obtain

$$\theta^\star = \arg\min_{\theta\in\Theta} \sum_{t=1}^{T} \Big(y_t - f_0(x_t) - f(x_t)^\top\theta - \mathbf{K}_t^\top \omega(\theta)\Big)^2 + \gamma\omega(\theta)^\top \mathbf{K}\omega(\theta).$$

Consider the centered output $y_t - f_0(x_t)$. Rewriting the summation and substituting (6) and (14), we obtain

$$\theta^\star = \arg\min_{\theta\in\Theta} \|Y_0 - F(x)\theta - \mathbf{K}\,\Psi(Y_0 - F(x)\theta)\|_2^2 + \gamma(Y_0 - F(x)\theta)^\top \Psi^\top \mathbf{K}\,\Psi(Y_0 - F(x)\theta) \tag{15}$$

where, we used the fact that (6a) corresponds to $\Gamma(\theta) = F(x)\theta$, and, according to (6b) and (14), $\omega(\theta) = (\mathbf{K} + \gamma\mathbf{I}_T)^{-1}(Y_0 - F(x)\theta) = \Psi(Y_0 - F(x)\theta)$. To simplify this expression further, consider

$$\Psi_1 \doteq \mathbf{I}_T - \mathbf{K}\,\Psi, \quad \Psi_2 \doteq \gamma\Psi^\top \mathbf{K}\,\Psi. \tag{16}$$

Substituting these definitions into (15), we write

$$\theta^\star = \arg\min_{\theta\in\Theta} \|\Psi_1(Y_0 - F(x)\theta)\|_2^2 + (Y_0 - F(x)\theta)^\top \Psi_2 (Y_0 - F(x)\theta). \tag{17}$$

Problem (17) can be recognized as a standard weighted least-squares (LS) problem, since the objective is quadratic in $Y_0 - F(x)\theta$, and can be written compactly as

$$\big(Y_0 - F(x)\theta\big)^\top \left(\Psi_1^\top\Psi_1 + \Psi_2\right)\big(Y_0 - F(x)\theta\big).$$

Therefore, the optimal solution has the closed form

$$\theta^\star = \left[F(x)^\top(\Psi_1^\top\Psi_1 + \Psi_2)F(x)\right]^{-1} F(x)^\top(\Psi_1^\top\Psi_1 + \Psi_2)Y_0. \tag{18}$$

To further simplify this expression, recall from the definition of Ψ in (14) that $(\mathbf{K} + \gamma\mathbf{I}_T)\Psi = \mathbf{I}_T$. Rearranging, this implies $\gamma\Psi = \mathbf{I}_T - \mathbf{K}\Psi$. Now, using this relation in (16), we obtain $\Psi_1 = \mathbf{I}_T - \mathbf{K}\Psi = \gamma\Psi$. Substituting into $\Psi_1^\top\Psi_1 + \Psi_2$ yields

$$\Psi_1^\top\Psi_1 + \Psi_2 = \gamma^2\Psi^\top\Psi + \gamma\Psi^\top\mathbf{K}\Psi = \gamma\Psi^\top(\gamma\mathbf{I}_T + \mathbf{K})\Psi$$

factoring out $\gamma\Psi^\top$ and Ψ. Hence, being $(\mathbf{K} + \gamma\mathbf{I}_T)\Psi = \mathbf{I}_T$ from (14), and Ψ symmetric, we conclude $\Psi_1^\top\Psi_1 + \Psi_2 = \gamma\Psi$. Substituting into (18) directly yields (13), with the scalar factor γ canceling out as it appears both inside the inverse and outside. This concludes the proof. □

Remark 3 (On matrix invertibility and system identifiability): It is worth noting that Ψ is always positive definite, being defined as the inverse of the matrix $(\mathbf{K} + \gamma\mathbf{I}_T)$. Indeed, $\mathbf{K}$ is guaranteed to be symmetric and at least positive semidefinite by Definition 1, and $\gamma\mathbf{I}_T \succ 0$ for any $\gamma > 0$. Consequently, the only requirement for the invertibility of $F(x)^\top\Psi F(x)$ is that $F(x)$ has full column rank, as assumed in Theorem 2. The full column rank condition requires that $T \geq n_\theta$ (i.e., at least as many data points as parameters) and that the regressor vectors $f(x_t)$ are linearly independent. This is equivalent to requiring that the input signal is persistently exciting. If $F(x)$ is not full column rank, the solution to (7a) is not unique, reflecting an identifiability issue due to insufficient excitation in the input signal. In this case, one can select the minimum-norm solution, obtained by replacing the inverse with the Moore–Penrose pseudoinverse, i.e., $\theta^\star = \left(F(x)^\top\Psi F(x)\right)^\dagger F(x)^\top\Psi Y_0$.

This result is particularly significant as it shows that, when the model is affine in θ, optimization problem (7a) becomes convex, ensuring a unique and efficiently computable solution. Indeed, the computation of the closed-form solution in (13) requires only standard matrix inversions. On the other hand, in the nonaffine case, computing the solution to (7a) requires iteratively computing the gradient and evaluating the kernel, thus leading to higher computational cost. Importantly, the structure in (12) is quite general, as it does not impose linearity with respect to x, but only in the parameters. Notably, many nonlinear (with respect to their inputs) systems can still be expressed in this form, making the framework and Theorem 2 broadly applicable. On the other hand, when affinity in the parameters does not hold, we can still tackle problem (7a) by means of nonlinear programming methods but due to the potential nonconvexity of the problem, the obtained solution may be local.

IV. Application to State-Space Systems

In the previous section, we considered a static input–output identification setting, where the goal was to estimate physical parameters $\bar{\theta}$ exploiting physical priors $f(x,\theta)$ and approximating an unknown function $\Delta(x)$ based on measured data pairs (x_t, y_t). In this section, we extend the kernel-based framework to dynamic settings characterized by not-fully measured states by introducing a state-smoothing approach to effectively address the associated challenges.

We consider a discrete-time system of the form

$$x_{t+1} = f(x_t, u_t, \bar{\theta}) + \Delta(x_t, u_t) + e_t \tag{19}$$

where $x_t \in \mathbb{R}^n$ denotes the state at time t, $u_t \in \mathbb{R}^{n_u}$ is the external measured input, and $e_t \in \mathbb{R}^n$ represents the process noise. The physics-based function $f(x_t, u_t, \theta)$ and the unmodeled dynamics $\Delta(x_t, u_t)$ are now vector-valued, each comprising n components, i.e., $f_i(x_t, u_t, \bar{\theta}_i)$ and $\Delta_i(x_t, u_t)$, $i = 1,\dots,n$. If all state variables are directly measurable, each parameter vector $\bar{\theta}_i$ and function Δ_i can be estimated using Theorem 1 directly. In this case, the state x_t serves both as the input—along with the measured input u_t—and as the measured output ($y_t = x_t$), allowing for a direct application of Theorem 1. However, this approach becomes infeasible when certain state components are not directly measurable. In such cases, system (19) is extended to incorporate also the output equation, i.e.,

$$\begin{aligned} x_{t+1} &= f(x_t, u_t, \bar{\theta}) + \Delta(x_t, u_t) + e_t \\ y_t &= g(x_t, u_t, \bar{\theta}) + w_t \end{aligned} \tag{20}$$

where w_t represents the measurement noise, and the state x_t is not directly accessible. However, this framework adds a further layer of complexity to the estimation procedure, which can be addressed by adopting two principal approaches: (i) multistep identification methods; or (ii) prior states estimation. As anticipated in the introduction, the optimization becomes challenging in multistep identification due to significant nonlinear dependencies among the parameters. Moreover, the recursive dependence of $\delta(x_t, u_t)$ within $f(x_t, u_t, \theta)$, and consequently θ, prevents a straightforward application of Theorem 1. To circumvent this issue, in the following, we focus on the second class of approaches, adopting nonlinear state smoothing techniques.

A. Nonlinear State Reconstruction

We consider the system described by (20), where the state variable $x_t \in \mathcal{X}$ evolves according to known physics-based functions $f : \mathcal{X} \times \mathbb{R}^{n_u} \times \Theta \to \mathcal{X}$, and $g : \mathcal{X} \times \mathbb{R}^{n_u} \times \Theta \to \mathbb{R}$, and an unknown term $\Delta : \mathcal{X} \times \mathbb{R}^{n_u} \to \mathcal{X}$, related to unmodeled dynamics. The goal of nonlinear state smoothing is to estimate a state trajectory $x_{0:T-1} \doteq \{x_0, \dots, x_{T-1}\}$ from a given dataset of measurements $\mathcal{D} = \{(u_0, y_0), \dots, (u_{T-1}, y_{T-1})\} \cup \{y_T\}$ and a nominal nonlinear model [18].

First, let us consider the known components of (20), which define the nominal model, i.e.,

$$x_{t+1} = f(x_t, u_t, \theta_0) + e_t$$
$$y_t = g(x_t, u_t, \theta_0) + w_t$$

where θ_0 represents the initial parameter estimate used for the state smoothing. This can correspond, for example, to an initial guess or to the central point in the parameter space Θ, which will be refined during the identification process. Moreover, without loss of generality, we assume that an initial estimate of the initial condition, denoted as $\hat{x}_0$, is available. This estimate can be seamlessly incorporated into the identification problem alongside θ if needed (see, e.g., [8]).

To perform the nonlinear state reconstruction, in this work, we employ a nonlinear state smoothing based on a two-step strategy. First, we apply a *forward filtering*, based on an UKF, for state estimation. Then, we employ a *backward smoothing* for refining the state estimates. However, we note that any state reconstruction approach can be employed in this last step. Therefore, we begin by imposing standard conditions ensuring that the state can be estimated and the physical parameters can be identified from the available measurements.

Assumption 1 (observability and identifiability): The system state x is observable along the trajectory induced by the applied input, and the physical parameters θ are identifiable [8], provided that the input is persistently exciting over the observation window.

Next, we detail the proposed two-step approach as applied to the considered state-space framework.

1) Forward filtering: The UKF aims to approximate the posterior state distribution using a set of sigma points, which are propagated through the nominal nonlinear system dynamics. Here, we assume that the process and measurement noise, e_t and w_t, can be characterized as zero-mean Gaussian with covariances P_e and P_w, respectively, i.e., $e_t \sim \mathcal{N}(0, P_e)$, $w_t \sim \mathcal{N}(0, P_w)$. The state estimate uncertainty, represented by P_t, is initialized as P_0 and updated recursively. All common variants of the UKF for discrete-time systems adhere to the same prediction–correction structure, though they may differ in specific formulations and weight definitions. See, e.g., [17] for details on the UKF and its implementation.

At the end of the forward filtering step, we obtain the filtered state sequence $\hat{x}_{1:T} \doteq \{\hat{x}_1, \dots, \hat{x}_T\}$ with the associated covariance matrices $P_{1:T} \doteq \{P_1, \dots, P_T\}$. These estimates serve as the input for the subsequent smoothing process.

2. Backward smoothing: The URTSS aims to obtain the final state estimates [18]. Given the filtered estimates, sigma points are generated and propagated through the model to compute predicted means, covariances, and cross-covariances. The smoother gain matrix is then used to iteratively refine the state and covariance estimates, running backward from time $t = T - 1$ to $t = 0$, yielding the final smoothed state sequence $\hat{x}^s_{0:T-1} \doteq \{\hat{x}^s_0, \dots, \hat{x}^s_{T-1}\}$ and covariances $P^s_{0:T-1} \doteq \{P^s_0, \dots, P^s_{T-1}\}$.

B. State-Space Reformulation

Once the sequence of unmeasured states has been reconstructed through state smoothing, i.e., $\hat{x}^s_{0:T-1}$, the problem effectively reduces to the one in (5), to which Theorem 1 is directly applicable. This can be done, for instance, by defining $z_t \doteq [\hat{x}^s_{t-1}, u_{t-1}, u_t]$ as the *new* input variable, and computing the predictions at time t, i.e., $\hat{y}_t(\theta, \delta)$ in (5), using the following prediction model:

$$\hat{y}_t = \xi(z_t, \theta) + \delta(z_t) \tag{21}$$

with $\xi(z_t) \doteq g(f(\hat{x}^s_{t-1}, u_{t-1}, \theta), u_t, \theta)$. In this formulation, $\delta(z_t)$ captures the discrepancies arising from unknown or unmodeled components in $f(x_t, u_t, \theta)$ (20), which influence the system evolution and are subsequently mapped to the output space by $g(x_t, u_t, \theta)$.

We note that, although alternative models exist, the selected formulation fully leverages the available physical priors by incorporating both $f(x_t, u_t, \theta)$ and $g(x_t, u_t, \theta)$. The proposed methodology enables the estimation of physical parameters θ and the approximation of the unknown component $\Delta(x_t, u_t)$ via a kernel-based approach, effectively embedding prior knowledge with data. This principled combination enhances both interpretability and accuracy. In addition, the nonlinear state smoother preserves the well-behaved properties of single-step identification while implicitly capturing multistep dependencies. This leads to improved accuracy in both single- and multistep settings, without requiring explicit multistep optimization.

V. Numerical Example

In this section, we illustrate the effectiveness of the proposed identification method on an academic example and on a cascade tank system (CTS) benchmark [24].

A. Academic Example

We first address a regression problem aiming to estimate the parameters $\bar{\theta}$ of a linear-in-parameter model using the kernel-based approach, and compare it with ordinary LS, the discrepancy modeling (DM) approach, and standard KRR (i.e., without embedding prior physical knowledge).

We generate a total of $T = 1000$ samples with input values $x \in [-2, 2]$ and outputs following a polynomial and sinusoidal relationship. The dataset is split into three parts: (i) 500 samples from $x \in [-1, 1]$, employed for training; (ii) 250 samples from $x \in [1, 2]$, used as a validation set for hyperparameter selection; and (iii) 250 samples from $x \in [-2, -1]$, reserved a as test set for performance evaluation. Relying on (12), we have

$$f_0(x) = 0,\ f(x)^\top = [1, x, u, x^2, u^2]$$
$$\Delta(x) = 0.7\sin(5x) + 0.5\cos(3x) + 0.4x^2 + 0.3x^3 - 0.2\sin(7x)\cos(2x) \tag{22}$$

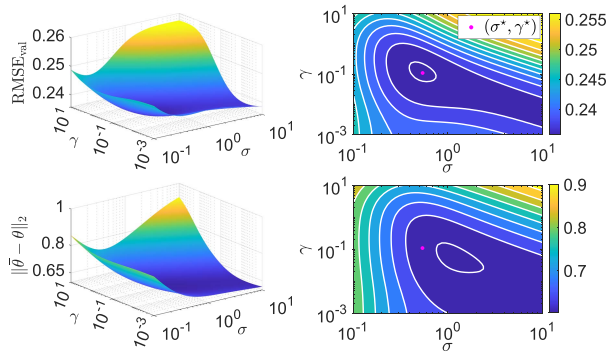

Fig. 1. Validation RMSE as a function of the kernel bandwidth σ and the regularization weight γ (log scale).

TABLE I
IDENTIFICATION PERFORMANCE WITH DIFFERENT METHODS

	θ_1 [-]	θ_2 [-]	θ_3 [-]	θ_4 [-]	θ_5 [-]	RMSE_{tst} [-]
True	2	3	4	1.5	−0.8	0.097
LS	2.55	3.03	4.32	0.80	−1.05	0.961
DM	2.52	3.03	4.32	0.81	−1.04	0.929
KRR	–	–	–	–	–	1.206
Proposed	**2.18**	**2.85**	**4.16**	**1.20**	**−0.83**	**0.343**

The bold values indicate the results achieved by the proposed kernel-based identification framework.

where $u = \sin(2\pi x) + 0.5\cos(3\pi x)$, e is a zero-mean Gaussian distribution with standard deviation 0.1, and $\bar{\theta} = [2, 3, 4, 1.5, -0.8]$. To improve the estimation accuracy in the presence of unknown nonlinearities $\Delta(x)$, we employ a Laplacian kernel with $\mathbf{K}$ entries as $\mathbf{K}_{ij} = \exp(-|x_i - x_j|/\sigma)$. The optimal hyperparameters σ and γ are selected via a validation-based grid search. Models are trained for different (σ, γ) pairs using the training set and evaluated by root mean square error (RMSE) on the validation set. A 50×50 logarithmic grid is explored with $\sigma \in [10^{-1}, 10^{1}]$ and $\gamma \in [10^{-3}, 10^{1}]$, yielding the optimal pair $(\sigma^\star, \gamma^\star) = (0.54, 0.11)$ that minimizes the validation RMSE.

The outcome of this procedure is shown in Fig. 1, which reports the validation RMSE (RMSE_{val}) and the parameter error ($|\bar{\theta} - \theta|_2$) for each tested (σ, γ) pair. Note that hyperparameters minimizing the RMSE do not necessarily coincide with those minimizing the parametric error. However, being the true parameter vector $\bar{\theta}$ unknown in practice, it cannot be directly exploited. Nevertheless, as illustrated in Fig. 1, the RMSE-based selection provides a reliable criterion, yielding parameter estimates close to the minimum parameter error. The impact of γ on the proposed identification process is as expected: small values cause higher RMSE due to kernel overfitting and reduced influence of physical priors, whereas large values yield biased parameter estimates by enforcing excessive adherence to the physical model and failing to capture unmodeled effects.

Once the hyperparameters are selected, the kernel-based estimator follows the formulation derived in Theorems 1 and 2, where the correction is introduced through a kernel, and the operator Ψ projects the observations into a feature space that captures unmodeled nonlinear effects. The results in Table I compare the proposed approach with the true system (for reference on noise level), LS, the DM method [5], and standard KRR without the parametric component (i.e., prior knowledge on $f_0(x)$, $f(x)$ is not exploited).[1] Results are compared in terms of obtained parameter estimates ($\theta_i, i = 1, \ldots, 5$), and RMSE on the test set (RMSE_{tst}).

TABLE II
STATISTICAL EVALUATION OVER 1000 MONTE CARLO EXPERIMENTS

	$\|\bar{\theta} - \hat{\theta}\|_2$ [-]	fit_{tst} [%]	RMSE_{tst} [-]
True	0 ± 0	96.07 ± 1.45	0.099 ± 0.0045
LS	0.966 ± 0.014	62.44 ± 13.83	0.953 ± 0.032
DM	0.966 ± 0.014	64.91 ± 13.34	0.891 ± 0.104
KRR	–	38.80 ± 12.16	1.281 ± 0.241
Proposed	$\mathbf{0.517 \pm 0.056}$	$\mathbf{78.74 \pm 7.67}$	$\mathbf{0.541 \pm 0.138}$

The bold values indicate the results achieved by the proposed kernel-based identification framework.

The estimated parameter vector $\theta^\star$, obtained from (13) with $(\sigma^\star, \gamma^\star)$, is notably closer to the true $\bar{\theta}$ than the ordinary LS estimate θ^{LS}, which ignores unknown nonlinearities. Moreover, the proposed kernel-based model achieves an RMSE of 0.343, compared to 0.961 for the LS model. The estimate θ^{LS} also corresponds to that obtained with the DM approach. In this two-step scheme, the physical parameters are first identified without accounting for $\Delta(x)$, and the residual discrepancy is then modeled separately using a Laplacian kernel correction. Although this reduces the RMSE to 0.929, it remains inferior to the proposed method, which jointly estimates the physical parameters and unmodeled dynamics, yielding both lower RMSE and less biased parameters. Notably, the performance of the standard KRR model confirms the importance of embedding prior physics. In fact, despite its flexibility, KRR does not estimate physical parameters and yields a substantially higher test RMSE, indicating overfitting to training data and limited generalization capability when used alone.

These results illustrate the benefits of using a kernel-based embedding approach. By incorporating the unmodeled effects into the estimation process, the kernel-based approach better represents the system dynamics, leading to significantly lower RMSE values and improved parameter estimates.

To evaluate the robustness and statistical significance of the results, a Monte Carlo analysis is performed over 1000 independent identification runs. In each trial, a system following (22) is generated with random noise realizations and true parameters obtained by perturbing the nominal values $[2, 3, 4, 1.5, -0.8]$ uniformly within $\pm 50\%$ of their magnitude. The proposed kernel-based method (with Laplacian kernel) is again compared with a standard LS estimator, a DM approach (with Laplacian kernel), and a pure KRR model. For each trial, the training, validation, and test sets are defined as before, and hyperparameters for the proposed, DM, and KRR methods are tuned using the same validation-based procedure.

Table II reports the average parameter estimation error $\|\bar{\theta} - \hat{\theta}\|_2$, the fit percentage (fit $\doteq 100(1 - \|y - \hat{y}\|_2/\|y - \bar{y}\|_2)$, with $\bar{y} = \frac{1}{T}\sum_{t=1}^{T} y_t$) on the test data, and the test RMSE with their corresponding standard deviations over the 1000 Monte Carlo runs. The results confirm the statistical consistency and robustness of the proposed approach, which achieves the smallest mean parameter error and the lowest test RMSE, with limited variability across trials.

B. Cascade Tank System Benchmark

In the second example, we test the proposed approach on a CTS benchmark [24] and compare it with other state-of-the-art methods.

[1] Same validation procedure to tune KRR and DM hyperparameters, yielding $(\sigma^\star = 0.1, \gamma^\star = 10^{-2})$ for DM and $(\sigma^\star = 10, \gamma^\star = 10^{-2})$ for KRR.

TABLE III
PERFORMANCE ANALYSIS ON ESTIMATION AND VALIDATION DATA

	Pred. (RMSE [V])		**Sim.** (RMSE [V], fit [%])	
	Train.	Val.	Train.	Val.
(23) **only**	0.06	0.06	0.37, 83.14	0.37, 82.46
(23) **+ kernel**	0.04	0.05	0.17, 92.11	0.18, 91.55

The discretized model of the CTS is given by

$$\begin{aligned} x_{1,t+1} &= x_{1,t} + T_s\left(-k_1\sqrt{x_{1,t}} + k_4 u_t + e_{1,t}\right) \\ x_{2,t+1} &= x_{2,t} + T_s\left(k_2\sqrt{x_{1,t}} - k_3\sqrt{x_{2,t}} + e_{2,t}\right) \\ y_t &= x_{2,t} + w_t \end{aligned} \tag{23}$$

with $u_t \in \mathbb{R}$ the input signal, $x_{1,k} \in \mathbb{R}$ and $x_{2,k} \in \mathbb{R}$ the state variables, $y_t \in \mathbb{R}$ the output, and $e_t \in \mathbb{R}^2$, $w_t \in \mathbb{R}$ the additive noise sources. The sampling time is set to $T_s = 4$ s. The system is characterized by four unknown physical constants, k_1, k_2, k_3, and k_4, which depend on the properties of the system and need to be estimated. Moreover, unmodeled dynamics, e.g., water overflow effect, are included in physical dynamics model (23). See [8] for further details. The training and validation datasets consist of $T = 1024$ input–output samples each. A 10% of the training data was reserved for hyperparameter tuning, carried out as in the previous example, while the original benchmark validation dataset was kept unchanged to ensure a fair comparison with existing results.

The goal is to estimate the dynamics of the system using only the available training data. For the identification, we employ the nonlinear state smoothing assuming that $f(x_t, u_t, \theta)$ and $g(x_t, u_t, \theta)$ are defined according to the discretized model (23) and selecting $P_e = 10^{-3} I_2$, $P_w = 10^{-2}$, $P_0 = 0.5 I_2$, $\theta_0 = [0.05, 0.05, 0.05, 0.05]^\top$, and $\hat{x}_0 = [y_0, y_0]^\top$. Moreover, we set the UKF weights according to the formulation in [17], that is $a = 2.74$, $w_0^m = 0.33$, $w_0^c = 2.33$, and $w_i^m = w_i^c = 0.67$, for $i = 1, \ldots, 2\,\mathrm{n}$. Then, we solve an optimization problem of form (5) for $\gamma = 0.1$. Specifically, once the smoothed trajectory of the unmeasured state $\hat{x}^s_{1,0:T-1}$ is computed, we define a predictive model of form (21) and we solve Problem (5) applying Theorem 1. Hence, considering (23) and selecting $z_t \doteq [\hat{x}^s_{1,t-1}, y_t, u_{t-1}]$, we define $\xi(z_t) \doteq y_t + T_s k_2\sqrt{\hat{x}^s_{1,t-1} + T_s(-k_1\sqrt{\hat{x}^s_{1,t-1}} + k_4 u_{t-1}) - T_s k_3\sqrt{y_t}}$. Thus, once we obtain $(\theta^\star, \delta^\star)$ as the solution of (5) according to Theorem 1, the optimal estimation model becomes $\hat{y}_{t+1} = \xi(z_t, \theta^\star) + \delta^\star(z_t)$, selecting the Gaussian kernel $\kappa(x, x') = \exp(-\|x - x'\|^2/2\sigma^2)$ with $\sigma = 11$.

To evaluate the effectiveness of the estimation algorithm, the RMSE is employed as the performance metric, as suggested in [24]. Table III presents the performance of our method compared with the solution obtained by solving (5) when no kernel is used to compensate for unmodeled dynamics in (23). The results are reported for the estimation and validation datasets, considering: (i) the *prediction* task, i.e., given z_t, we estimate y_{t+1}; and (ii) the *simulation* task, i.e., we recursively estimate y_{t+1} by defining z_t with $\hat{y}_t$ (the previous estimate of y_t). Moreover, for the simulation results, we also report the fit values in Table III. These first results highlight the significant reduction in the RMSE achieved through kernel embedding, particularly in the simulation setting. Notably, despite the optimization being performed minimizing the prediction error, the joint use of the kernel-based approach and the smoother substantially improves the multistep simulation accuracy. This improvement is also reflected in the results reported in Table IV, where we compare the simulation performance for the validation data with other state-of-the-art approaches from literature, including approaches relying on predefined dictionary of basis functions, e.g., [8].

TABLE IV
METHODS COMPARISON (SIMULATION RMSE ON VALIDATION DATA)

Method	**RMSE** [V]	**Method**	**RMSE** [V]
Svensson et al. [3]	0.45	PWARX [25]	0.35
Volt.FB [11]	0.39	SED-MPK [12]	0.48
INN [2]	0.41	PNLSS-I [26]	0.45
NLSS2 [26]	0.34	Donati et al. [8]	0.26
Proposed	**0.178**		

For completeness, we report the identified parameters. The smoothed initial condition is $\hat{x}^s_0 = [4.78, 5.20]$ whereas, for $\hat{k}$, we have: 1) with no kernel, $\hat{k} = [-0.01, 0.05, 0.06, 0.01]$; and 2) with kernel, $\hat{k} = [0.08, 0.05, 0.05, 0.06]$.

VI. CONCLUSION

This work introduced a kernel-based framework for physics-informed nonlinear system identification, effectively embedding kernel methods while preserving physical model interpretability. To tackle the case of unmeasured states, we incorporated a nonlinear state smoothing. The numerical results confirm the effectiveness of the proposed approach in finding meaningful parametric models with compensating unstructured components.

Future investigations will explore advanced strategies for exploiting state smoothing with identification, analyzing the role of model uncertainty. Potential directions include weighted fitting based on the smoother covariance P^s_t and the iterative refinement of the filtering model. Additional extensions will focus on improving the hyperparameter selection procedure by, e.g., incorporating external physical information or constraints on the parameters.

ACKNOWLEDGMENT

The authors would like to thank T. Alamo for his valuable suggestion to incorporate kernel-models in our framework.

REFERENCES

[1] L. Ljung, "Perspectives on system identification," *Annu. Rev. Control*, vol. 34, no. 1, pp. 1–12, 2010.
[2] B. Mavkov, M. Forgione, and D. Piga, "Integrated neural networks for nonlinear continuous-time system identification," *IEEE Control Syst. Lett.*, vol. 4, no. 4, pp. 851–856, Oct. 2020.
[3] A. Svensson and T. B. Schön, "A flexible state–space model for learning nonlinear dynamical systems," *Automatica*, vol. 80, pp. 189–199, 2017.
[4] W. Quaghebeur, I. Nopens, and B. De Baets, "Incorporating unmodeled dynamics into first-principles models through machine learning," *IEEE Access*, vol. 9, pp. 22014–22022, 2021.
[5] K. Kaheman, E. Kaiser, B. Strom, J. N. Kutz, and S. L. Brunton, "Learning discrepancy models from experimental data," in *Proc. 58th IEEE Conf. Decis. Control*, 2019, pp. 7389–7396.
[6] M. Forgione, A. Muni, D. Piga, and M. Gallieri, "On the adaptation of recurrent neural networks for system identification," *Automatica*, vol. 155, 2023, Art. no. 111092.
[7] Y.-C. Zhu, P. Gardner, D. J. Wagg, R. J. Barthorpe, E. J. Cross, and R. Fuentes, "Robust equation discovery considering model discrepancy: A sparse Bayesian and Gaussian process approach," *Mech. Syst. Signal Process.*, vol. 168, 2022, Art. no. 108717.
[8] C. Donati, M. Mammarella, F. Dabbene, C. Novara, and C. M. Lagoa, "Combining off-white and sparse black models in multi-step physics-based systems identification," *Automatica*, vol. 179, 2025, Art. no. 112409.
[9] A. Carè, R. Carli, A. Dalla Libera, D. Romeres, and G. Pillonetto, "Kernel methods and Gaussian processes for system identification and control: A road map on regularized kernel-based learning for control," *IEEE Control Syst. Mag.*, vol. 43, no. 5, pp. 69–110, Oct. 2023.
[10] G. Pillonetto and G. De Nicolao, "A new kernel-based approach for linear system identification," *Automatica*, vol. 46, no. 1, pp. 81–93, 2010.

[11] M. Schoukens and F. G. Scheiwe, "Modeling nonlinear systems using a Volterra feedback model," in *Proc. Workshop Nonlinear Syst. Identification Benchmarks*, 2016, p. 42.
[12] A. Dalla Libera, R. Carli, and G. Pillonetto, "Kernel-based methods for volterra series identification," *Automatica*, vol. 129, 2021, Art. no. 109686.
[13] B. Schölkopf, R. Herbrich, and A. J. Smola, "A generalized representer theorem," in *Proc. Int. Conf. Comput. Learn. Theory*, 2001, pp. 416–426.
[14] A. H. Ribeiro, K. Tiels, J. Umenberger, T. B. Schön, and L. A. Aguirre, "On the smoothness of nonlinear system identification," *Automatica*, vol. 121, 2020, Art. no. 109158.
[15] M. Farina and L. Piroddi, "Simulation error minimization identification based on multi-stage prediction," *Int. J. Adaptive Control Signal Process.*, vol. 25, no. 5, pp. 389–406, 2011.
[16] T. B. Schön, A. Wills, and B. Ninness, "System identification of nonlinear state-space models," *Automatica*, vol. 47, no. 1, pp. 39–49, 2011.
[17] E. A. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proc. IEEE Adaptive Syst. Signal Process., Commun., Control Symp.*, 2000, pp. 153–158.
[18] S. Särkkä, "Unscented Rauch–Tung–Striebel smoother," *IEEE Trans. Autom. Control*, vol. 53,no. 3, pp. 845–849, Apr. 2008.
[19] N. Aronszajn, "Theory of reproducing kernels," *Trans. Amer. Math. Soc.*, vol. 68, no. 3, pp. 337–404.
[20] C. Saunders, A. Gammerman, and V. Vovk, "Ridge regression learning algorithm in dual variables," in *Proc. 15th Int. Conf. Mach. Learn.*, 1998, pp. 515–521.
[21] G. Wahba, *Spline Models for Observational Data*. Philadelphia, PA, USA: SIAM, 1990.
[22] D. Ruppert, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, NY, USA: Taylor & Francis, 2004.
[23] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statist. Comput.*, vol. 14, pp. 199–222, 2004.
[24] M. Schoukens, P. Mattson, T. Wigren, and J.-P. Noël, "Cascaded tanks benchmark combining soft and hard nonlinearities," in *Proc. Workshop Nonlinear Syst. Identification Benchmarks*, 2016, pp. 20–23.
[25] P. Mattsson, D. Zachariah, and P. Stoica, "Identification of cascade water tanks using a PWARX model," *Mech. Syst. Signal Process.*, vol. 106, pp. 40–48, 2018.
[26] R. Relan, K. Tiels, A. Marconato, and J. Schoukens, "An unstructured flexible nonlinear model for the cascaded water-tanks benchmark," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 452–457, 2017.

Open Access funding provided by 'Politecnico di Torino' within the CRUI CARE Agreement