Doctoral Dissertation
Doctoral Program in Electrical, Electronics and Communications Engineering
(XXXVIII cycle)

Physics-informed identification of nonlinear dynamical systems

A unified framework combining domain knowledge and data-driven sparse modeling

Cesare Donati

* * * * *

Supervisors
Prof. Carlo Novara
Dr. Fabrizio Dabbene
Prof. Giuseppe C. Calafiore

Doctoral Examination Committee:
Prof. Simone Formentin, Referee, Politecnico di Milano, Italy
Prof. Laura Giarrè, Referee, University of Modena and Reggio Emilia, Italy
Prof. Diego Regruto, Politecnico di Torino, Italy

Politecnico di Torino
2026

I hereby declare that, the contents and organisation of this dissertation constitute my own original work and does not compromise in any way the rights of third parties, including those relating to the security of personal data.

Cesare Donati
Turin, 2026

Summary

This thesis investigates the problem of physics-informed identification of nonlinear dynamical systems, proposing a unified framework that integrates partial physical knowledge with data-driven models to achieve interpretable, reliable, and scalable identification. The research focuses on the explicit combination of known physical equations with a corrective black-box component that accounts for unmodeled dynamics. In this context, the proposed methodology builds upon the notion of off-white models. These correspond to grey-box representations where part of the model structure is derived from first principles and physics, while unknown parameters must be identified from data.

The first contribution is the formulation of a *multi-step identification framework* in which the cumulative prediction error over extended horizons is minimized. Unlike classical single-step identification approaches, which often yield models that perform well only for short-term predictions, the proposed multi-step formulation provides consistency and reliability over longer horizons, a critical aspect in control and forecasting applications.

A second contribution is the introduction of a *sparse black-box augmentation* strategy that complements the known physical dynamics. The black-box component, initially expressed as a sparse combination of basis functions, captures only the missing dynamics required to model discrepancies. This formulation enables the joint estimation of physical parameters and residual dynamics, preventing the bias typically introduced by alternative identification procedures. Theoretical results are established, providing explicit bounds on the parametric estimation error and conditions for maximum sparsity recovery, thus ensuring both interpretability and accuracy.

The framework is then extended to address the challenge of *non-uniform observations*, a frequent feature of real-world data due to missing samples, multiple experimental runs, or temporally aggregated measurements. New formulations and bounds are derived to quantify the effect of these irregularities on parameter estimation, demonstrating the robustness of the method across heterogeneous data collection settings.

To overcome the limitations associated with the manual selection of basis functions, a *kernel-based extension* is then introduced. By embedding the model residuals in a reproducing kernel Hilbert space, the framework enables a nonparametric, data-adaptive

correction of the nominal physical model. This extension preserves the interpretability of the physical parameters while removing the dependence on predefined function dictionaries. The kernel-based formulation is further developed in the state-space setting, integrating kernel approximations with state reconstruction techniques such as unscented Kalman filtering and smoothing, resulting in enhanced predictive and simulation accuracy.

The proposed methods are empirically validated through a range of representative benchmarks and real-world-inspired case studies, including spacecraft inertia identification, cascade tank systems, continuous stirred-tank reactors, and ecological population dynamics, demonstrating robustness, interpretability, and improved long-horizon predictive performance under realistic data conditions.

Altogether, this thesis contributes a general and theoretically grounded approach to physics-informed identification, spanning sparse and kernel-based regularization, multi-step optimization, and non-uniform data handling. The proposed framework bridges the gap between traditional system identification and modern machine learning paradigms, establishing new foundations for a structured and interpretable modeling of nonlinear dynamical systems.

Acknowledgements

I would like to express my sincere gratitude to all those who have guided and inspired me throughout this Ph.D. journey, for their continuous mentorship, support, and encouragement.

Dr. Fabrizio Dabbene has been a mentor since the very beginning, from my Master's thesis to this dissertation. His ability to guide research while fostering independence has been essential to my growth as a researcher. I am deeply grateful for his constant support, his scientific rigor, and the inspiration he provided to approach problems with curiosity and perseverance.

Prof. Carlo Novara, who introduced me to the field of system identification, from its foundations to the advanced concepts presented in this thesis. His technical rigor and methodological clarity have had a decisive impact on my scientific development. His guidance has been invaluable in addressing complex problems and in shaping the future directions of my research.

Prof. Constantino Lagoa, who has been an essential mentor during the past three years. Our weekly meetings were technically invaluable and helped shape my approach to research and academic work. His expertise and vision have been a constant source of inspiration. The period spent at Penn State University under his supervision was one of the most rewarding experiences of my Ph.D., leaving me with insights and lessons of lasting scientific and personal value.

Dr. Martina Mammarella, who has played a fundamental role in my scientific growth during these years. Her rigor, precision, and perseverance have been an example to follow and a constant source of motivation. She has taught me the importance of pursuing ambitious goals with determination and clarity, even through challenging phases. Her guidance and mentorship have been invaluable to my development as a researcher.

Prof. Giuseppe Calafiore for the stimulating discussions and exchanges of ideas, which enriched my perspective on research and helped me expand my knowledge beyond the strict boundaries of system identification.

Finally, I would like to thank all the colleagues, collaborators, and friends I have met along this academic path, whose support, insights, and shared experiences have contributed to my professional and personal growth.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview and motivations

Modeling real-world physical systems is a fundamental task in engineering: inferring models from observations and studying their properties is, in essence, what science is about [1]. From the early development of classical mechanics to contemporary advances in data-driven modeling, the effort to understand, predict, and control the evolution of dynamical systems has remained at the core of scientific discovery and technological innovation. Within this broad context, *system identification* has emerged as the discipline of building mathematical models of dynamical systems from measured data, linking observed signals into structured models that can be analyzed, simulated, controlled, and used for decision-making [1], [2].

Dynamical systems are indeed widespread across nearly all domains of engineering and applied sciences. From aerospace [3], automotive [4], and energy systems [5], [6] to biological networks [7], climate models [8], and financial markets [9], accurate models are indispensable for tasks such as prediction, monitoring, diagnosis, and control. In modern technology, the ability to capture the behavior of complex processes through reliable models translates directly into improved safety, efficiency, and sustainability. For instance, in control design, a precise model is often the enabling step for implementing advanced strategies such as model predictive control [10]; in fault detection, identification techniques allow one to distinguish between nominal and anomalous behaviors (see, e.g., [11]); in simulation and digital twins, identified models provide computationally efficient surrogates of reality [12].

Thus, system identification plays a central role in bridging the gap between empirical data and theoretical modeling, offering a systematic methodology for extracting knowledge from observations and embedding it into mathematical structures. Moreover, its relevance has steadily increased with the growing availability of high-resolution sensors and large datasets, which open unprecedented opportunities while simultaneously raising new challenges in terms of scalability, robustness, and interpretability. In

this sense, identification is not merely a technical tool but rather a key component contributing to the integration of physics-based and data-driven approaches in the study of dynamical systems, as will be explored in this thesis.

1.1.1 The modeling spectrum

Traditionally, modeling real-world physical systems has relied on linear dynamical models characterized by well-defined parametric structures, such as AutoRegressive with eXogenous inputs (ARX), AutoRegressive Moving Average with eXogenous inputs (ARMAX), or linear state-space representations [1]. These methods have been extensively studied over the past decades, leading to a mature and well-established framework, together with efficient numerical algorithms for parameter estimation and model validation.

While these approaches are well understood, they often fall short in capturing the behaviors exhibited by real-world systems [13], [14]. Indeed, the majority of the systems encountered in modern engineering applications exhibit dynamical behaviors that may be too complex to be captured by linear relationships. As a result, the field of nonlinear system identification has experienced significant growth, and many approaches have been studied aiming at identifying nonlinear dynamical models from collected data. Although significant developments have been proposed [13], the problem remains largely open and continues to pose major theoretical and practical challenges [15].

Existing approaches to nonlinear systems identification can be classified into two main groups. On one side, methods arising from *basic principles*, in which the model is directly derived from the knowledge of the physical laws governing the observed system and of the relationships between the subsystems composing it. In particular, when the values of the physical parameters entering into the systems (e.g., masses or inertia of a spacecraft, or reaction coefficients of a chemical process) are themselves derived from separate, dedicated measurements or are somehow known, one is considering a so-called *white-box* model. This is, however, a rather extreme situation; more generally, some parameters require identification from data, and one enters the broad family of so-called *grey-box* models. Adopting the classification proposed by L. Ljung in [13], in this thesis, we will refer to this particular class of models as *off-white*.

On the other side of the spectrum, we have the *black-box* models, which aim at describing the system's dynamics using generic linear parameterizations [16], [17] or families of universal approximators [18]. This second class of models has gained increasing popularity for their adaptability to a wide range of systems with minimal (or no) prior information about the underlying physical processes. Notably, the majority of published works focus on some variation of the black-box model approach (see, e.g., [18] and references therein), while only few methodological works specifically discuss off-white methods (see for instance [19]). However, despite the desirable properties of black-box models, these show several limitations as, for instance, lack of interpretability, poor consistency with physical properties, and absence of shared guidelines for

selecting basis functions and model order.

To deal with these issues, the community has adopted and adapted solutions from machine learning and artificial intelligence literature. On the one side, nonparametric approaches [20], [21], mainly kernel-based methods [22], [23], [24], have gained popularity for their ability to capture a large diversity of nonlinear behaviors without requiring complicated choices of basis functions. On the other side, techniques based on neural networks (NN), e.g., [25], have been proposed, showing a remarkable implementation ease and capability of recovering long-term behaviors. The number of works using different forms of NN-based system identification is steadily growing, and they are fast becoming the go-to solution in several application fields. A growing number of works, for instance, focus on specialized architectures for dynamical modeling, including recurrent neural networks (RNNs), long short-term memory (LSTM) networks, gated recurrent units (GRUs), and echo state networks (ESNs) [26], each designed to capture temporal-dependencies, handle long-range correlations, and spatial features, respectively, thereby enhancing the modeling of complex and dynamic systems. However, lately, these techniques have also revealed their main limitations, i.e., the need for a large amount of high-quality training data and the difficulty of capturing some inherent physical phenomena at the core of such data. The proposed solution to these two drawbacks has been the same: to devise ways to "bring the physics back" into the model. This led to the exponential growth of the family of physics-informed NNs (PINNs) [27]. The main feature of PINNs lies in exactly incorporating physical information through either physics-based loss functions or structural modifications, ensuring physical consistency between inputs and outputs [27].

In parallel, the inherent opacity of these purely black-box models has encouraged significant research into explainable AI (XAI). Contemporary efforts typically focus on post-hoc interpretation of black-box models, utilizing techniques such as LIME or SHAP [28]. For instance, in the context of data-driven predictive controllers, recent studies [29] have proposed explainability frameworks allowing to design feedback loops from data preserving prior system properties.

However, despite their promising capabilities, both PINNs and XAI methods remain fundamentally tied to black-box and neural network structures. In contrast, alternative perspectives embedding physical principles through different modeling paradigms may offer significant advantages in terms of computational efficiency, intrinsic interpretability, transparency by design, and robustness. In this context, a central challenge in system identification is noted by Ljung in [13]: we have to find descriptions that are flexible enough to cover many relevant nonlinear phenomena, at the same time as they allow inclusion of physical insight in order not to be too flexible.

These considerations underscore the need for efficient and reliable identification frameworks that can effectively balance physics-based structure with data-driven flexibility, enabling accurate and interpretable modeling even under challenging conditions such as incomplete, noisy, or poor data.

1.1.2 Performance criteria and optimization

Beyond model structure, another crucial aspect is the choice of the performance criterion, which determines how models are evaluated and optimized. Within this context, researchers have emphasized the importance of multi-step identification frameworks, moving from traditional single-step prediction to more demanding simulation-based criteria (see, e.g., [30], [31] and references therein). Traditional system identification techniques typically adopted a single-step prediction error minimization (PEM) perspective. The model is applied once at each time step to predict the system's next state or output (one-step-ahead). Then, the model parameters are identified by minimizing the differences between the measured outputs and the one-step-ahead predictions. For linear-in-parameter models, this formulation yields convex optimization problems that are well understood and can be solved efficiently with guaranteed convergence to a global optimum. However, while such an approach can be effective for short-term predictions, it often lacks accuracy when the goal is to simulate or control the system over extended time intervals [32], [33]. Moreover, in the case of state-space models, it relies on the availability of accurate state measurements, which are often not accessible. Unlike static regression models, a state-space formulation accounts for the evolution of hidden states, requiring estimating both system parameters and unmeasured state trajectories. As a consequence, when the system states are latent, the single-step formulation becomes impractical.

In contrast, a multi-step identification framework requires the recursive propagation of the predicted state over a time horizon, using the candidate model at each time step. This approach provides a more stringent and informative evaluation of model accuracy, particularly for control-oriented tasks and long-term forecasting applications. Additionally, it is naturally suited to state-space models in which the system states are not directly measurable, as it relies on simulating the full system trajectory rather than fitting predictions step-by-step to measured states. Clearly, the identification of such types of models may easily destroy nice convexity properties of the associated cost functions, thus leading to hard optimization problems [31]. This is primarily due to the nonlinear interactions between parameters, which become increasingly complex as predictions propagate over longer horizons.

Within this context, developing efficient and reliable optimization methods for multi-step identification remains a key research challenge, where advanced first-order optimization methods emerge as a promising solution. Such methods, which under suitable conditions are guaranteed to converge to a solution (in general sub-optimal), leverage gradient information to iteratively update decision variables [34] and have recently gained popularity for their ability to tackle large-scale and complex problems [35], [36]. Indeed, their effectiveness in solving non-convex problems remains one of the unresolved mysteries, contributing to the success of deep learning across numerous applications (see [37] and references therein), and makes them an appealing choice for tackling the optimization challenges presented by complex identification tasks.

1.1.3 Facing the real world: non-uniform observations

While the choice of performance criteria and optimization methods defines the way we extract information from data, the quality, completeness, and structure of the data themselves play an equally critical role. While in simulation environments data are typically dense and uniformly sampled, this condition rarely holds in real-world scenarios. In practice, measurements are often affected by irregular sampling or specific observability conditions, leading to datasets that are incomplete, fragmented or aggregated. This makes real-world data a common source of uncertainty in practical applications (see, e.g., [38], [39], and references therein) rendering conventional identification methods, which assume regularly sampled data, ineffective. This motivates the need of identification frameworks able to handle these situations, adopting techniques capable of treating non-uniform observations while ensuring consistency despite experimental variability.

The term *non-uniform observations* refers to data scenarios that may include: (i) missing measurements, (ii) multiple runs, i.e., repeated simulations of the system with different initial conditions, or (iii) aggregated outputs, where only averaged or accumulated system outputs over a time window are available, instead of individual readings. For example, sensor failures or varying sampling rates can create data gaps, while multiple runs may result from different experimental setups or varying external conditions [40]. On the other hand, aggregated outputs commonly arise in fields where continuous sampling is impractical. In such cases, monitoring the evolution of certain quantities can involve sampling measurements at extended intervals, providing only average values or accumulated information over these periods. In atmospheric or meteorological modeling, for example, weather stations record average temperature, humidity, or precipitation levels over several hours or days rather than continuously [41], [42]. In ecological and biological studies, averaged samples are used to study long-term ecological changes [43] or to improve the robustness of statistical analyses and provide a better understanding of population variability [44], [45]. In economics and finance, data such as gross domestic product, growth rates, or quarterly earnings are typically collected at extended intervals to provide a broader view of the economic trends and financial health over time (see, e.g., [46]). Similarly, in chemical industries, processes like the simulated moving bed involve the collection of samples at extended intervals due to a time-consuming and costly analysis [47].

Within this context, the majority of the literature focuses on system identification techniques under the assumption of missing data and multiple runs, i.e., measurements are either sporadically absent or completely missing for certain time steps. For example, an expectation maximization-based strategy is presented in [48] for data-driven identification with missing output observations. In this work, model parameters and missing observations are simultaneously and iteratively estimated using linear state-space models. In [49], several reconstruction methods for ARX models with missing

measurements were compared, including Kalman filtering, maximum likelihood estimation, and iterative reconstruction. All these methods often rely on the exploitation of specific model structures, such as ARX or linear state-space models, which may not generalize well to a large variety of real systems. Furthermore, with this class of approaches, the computational cost for reconstructing missing data becomes increasingly expensive as the amount of missing data grows.

Another class of approaches dealing with missing measurements leverages nuclear norm subspace identification methods, as in [47], [50], [51], where a convex optimization problem is formulated to estimate, in one step, both missing data and model parameters. Despite this class of techniques demonstrates robustness in handling incomplete dataset, their reliance on linear state-space models may jeopardize the model identification if the systems governing equations are highly nonlinear. Alternatively, an expectation-maximization algorithm that employs a particle filter and a particle smoother is employed in [52] and [53] for the identification of a nonlinear black box model under missing observations. Analogously, solutions based on black-box neural networks are proposed in [54] and [55] to effectively handle missing observations while identifying a system. Nonetheless, the effectiveness of black-box methods in modeling complex systems is limited by the lack of interpretability [13], [56]. This drawback compromises the possibility to rely on, e.g., prior knowledge of the model structure and physical constraints to compensate for the information lost due to missing measurements. Other recent approaches to missing data recovery include statistical and graph-based methods, such as kernel-based fault detection [57] and spatio-temporal graph convolutional networks [58], which have shown good performance in their respective domains but are not directly tailored to dynamic system identification tasks involving physical priors.

On the other hand, to the best of our knowledge, the specific challenge of aggregated outputs has not been investigated in the existing literature, leaving a notable gap in addressing the challenges posed by this type of observation data. This is a critical issue, as aggregation smooths short-term fluctuations, masks underlying dynamics, and distorts high-frequency components, leading to information loss between observation points.

1.2 Setting and contributions

The present thesis has been motivated by the fact that, although there have been significant advances in the area of system identification, there is still a need to develop identification algorithms that can: (i) leverage all the information available on a system, such as partial parametric description, measurements, and available information on intrinsic properties of the system, while remaining flexible enough to capture unmodeled

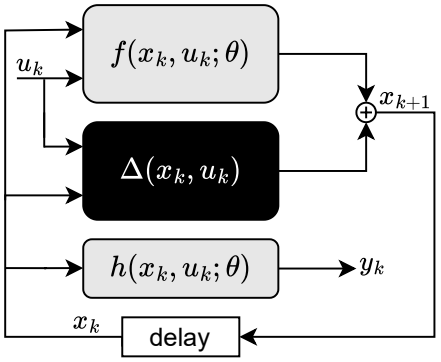


Figure 1.1: Dynamical system of interest.

nonlinear phenomena, (ii) deliver reliable estimates over extended time horizons, providing multi-step error minimization guarantees, and (iii) handle non-uniform observation. Clearly, such type of identification leads to challenging optimization problems, and thus motivates an additional parallel requirement: (iv) exploit recent advances in first-order optimization, at the core of the success of NN-based methods.

With the above objectives in mind, we take a substantial step towards using off-white models and propose an efficient framework with a threefold goal: (i) *physical interpretability* – we aim at deriving models whose parameters have a physical meaning and whose values are as close as possible to the "real" ones; (ii) *multi-step horizon* – the derived model should ensure more accurate long-term predictions, minimizing a multi-step prediction cost, to improve the reliability of the identified model; (iii) *flexibility to data* – the framework should be adaptable to different non-uniform observation conditions, including missing data, multiple runs, and aggregated measurements.

Within this thesis, the dynamical system of interest is assumed to be of the form depicted in Fig. 1.1. Specifically, we consider a system described by an off-white model, given by known nonlinear equations derived from physical principles, alongside discrepancies arising from, e.g., modeling errors, uncertainties, and perturbations that can affect parameter accuracy [59], [60]. This formulation is general enough to describe a broad class of nonlinear, time-invariant systems commonly encountered in engineering applications. Indeed, in many practical settings, ranging from mechanical and electrical systems to energy networks, environmental processes, and biological systems, a partial model of the underlying dynamics is often available. Such models typically arise from first-principles derivations or empirical laws and capture essential physical behaviors. However, these physical models are rarely perfect: they may neglect frictional effects, fluid turbulence, actuator delays, structural nonlinearities, or other

context-specific phenomena. These neglected dynamics, whether due to modeling simplifications, unmodeled coupling effects, sensor limitations, or environmental perturbations, can be represented compactly by the additive term Δ.

To effectively address these discrepancies while preserving physical interpretability and promote adherence to the true system parameters, we propose a mixed modeling approach that combines the known off-white equations with a sparse black-box component designed to compensate for unmodeled dynamics, initially defined as a combination of basis functions. In this framework, the model is recursively propagated over the entire prediction horizon, and the cumulative multi-step prediction error is minimized. Furthermore, the multi-step cost function is enriched with dedicated penalty terms that enforce relevant physical properties, such as passivity, monotonicity, or stability [61]. The resulting optimization problem is formulated in a general and flexible manner, enabling the framework to handle non-uniform data such as missing values, multiple experimental runs, and aggregated measurements, thus ensuring broad applicability to practical identification scenarios.

By coupling partial physical knowledge with data-driven models flexibility, the proposed formulation provably enhances both accuracy and reliability, improving output estimation and yielding more consistent estimates of the physical parameters. To this end, we focus on finding the sparsest coefficient vector selecting the basis functions to obtain a compensation term that only deals with the unmodeled dynamics when necessary and it is easily "interpretable". Indeed, non-sparse coefficient vectors might also capture part of the physical model's dynamics, resulting in predictions aligned with the measurements but with inaccurate physical parameter estimates.

It is important to distinguish the specific perspective adopted in this thesis from classical grey-box modeling approaches. In [13] grey-box models are defined as those where the model set is "shrunk as much as possible using physical insights". While this reduction in model complexity is beneficial, it introduces a bias error if the physical structure does not perfectly match reality. In standard grey-box identification, indeed, the model structure is typically assumed to be fully known and correct, with the identification task limited to estimating unknown physical parameters. In contrast, the framework proposed here explicitly acknowledges that the physical model is often an approximation; thus, it addresses not just parametric uncertainty, but also structural model-plant mismatch via an additive corrective term. Furthermore, unlike many hybrid or physics-guided machine learning approaches (e.g., PINNs) that often embed physical constraints into opaque black-box structures, this work prioritizes interpretability.

Similar model augmentation strategies have been investigated in the literature. For instance, [62] expands physics-based models via weighted regularization, while [63] employs the SINDy algorithm to capture discrepancies between simplified models and measurement data through sparsity-promoting techniques. However, a key assumption in both approaches is that the physical parameters are known a priori. In contrast, the focus of this thesis is on the joint identification of physical parameters and unmodeled dynamics, where the latter are captured through a black-box augmentation term.

Along the same lines, the work in [59] proposes to integrate machine learning components into first-principles models using neural networks. While effective in certain scenarios, this approach lacks explicit regularization mechanisms, which reduces the interpretability of the estimated physical parameters and increases the risk of overfitting. Moreover, no accompanying theoretical guarantees are provided, limiting the possibility of assessing the reliability of the obtained models. In contrast, our framework incorporates dedicated regularization to enforce sparsity and enhance interpretability, and is complemented by a theoretical analysis that establishes estimation bounds and sparsity recovery guarantees.

Within this context, some other 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 [64], [65]. 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. Such term, therefore, 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 this thesis. By modeling unmodeled dynamics explicitly from the beginning and estimating both the physical parameters and correction terms simultaneously, the interference between the two is minimized, leading to a more accurate and reliable identification process that prevents biased parameter estimates.

As introduced, as a starting point, we consider a black-box modeling approach based on the selection of basis functions. While this strategy proves effective in capturing nonlinear dynamics, it inherently relies on the availability of a well-chosen set of functions capable of representing the unmodeled effects. This requirement poses a significant limitation, as the performance of the approach is highly sensitive to the expressiveness and appropriateness of the chosen basis set [66]. Within this thesis, we also fill this gap by proposing a novel framework extension that integrates kernel methods with available physics-based models. Kernel methods [22], [66] 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. These methods are widely used in the context of input-output system identification (see, e.g., [23], [24]), 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.

The approach proposed in this thesis ultimately leverages kernel-based function approximation to systematically compensate for unmodeled dynamics while maintaining the interpretability of the physical part of the model. Overall, the framework is developed progressively within the thesis: we begin with a basis-function approach to capture unmodeled effects in a sparse and interpretable manner, and then extend it to a kernel-based formulation that adapts directly to the data. By exploiting the representer

theorem [67], the final framework provides a regularized, data-driven functional approximation mechanism that removes the need for manually selecting basis functions, while ensuring both interpretability and predictive accuracy through the embedded physical structure. Moreover, to accommodate a wider range of dynamical systems [68], the kernel-based approach, typically suited for traditional input-output models, is formulated in the general state-space setting depicted in Fig. 1.1, extending its applicability. This construction, from interpretable sparse bases to flexible kernel methods, reflects the central theme of the thesis: building a unifying, physically consistent, and flexible identification framework.

1.3 Thesis outline

The thesis is organized into six chapters, followed by three appendices, each addressing a specific aspect of the proposed identification framework and its theoretical and practical developments.

- Chapter 2 presents the core identification framework proposed in the thesis. It formalizes the mixed physics-data model structure, consisting of a known off-white component and a sparse black-box term that compensates for unmodeled dynamics. The chapter defines the multi-step cost function incorporating physical penalties and sparsity regularization and formulates the corresponding optimization problem. The approach is illustrated through two case studies: spacecraft inertia identification and the cascade tank system benchmark. The material in this chapter has appeared in [60], [69].
- Chapter 3 establishes theoretical results that characterize the estimation properties of the proposed framework. It first derives an explicit bound on the parameter estimation error, then studies conditions for maximum sparsity recovery in both single-step and multi-step settings. These results extend existing sparsity theorems to nonlinear, multi-step identification problems and analyze the optimality of the recovered physical parameters. The chapter concludes with an academic example illustrating the theoretical results. This chapter is based on the results published in [69].
- Chapter 4 extends the proposed framework to deal with real-world data scenarios characterized by non-uniform observations. It introduces formulations capable of handling missing observations, multiple runs, and temporally aggregated measurements within the same unified setting. Theoretical bounds are derived for the parameter estimation error under missing or aggregated data. The chapter concludes with practical case studies, including a continuous stirred-tank reactor system and an ecological predator-prey model, demonstrating robustness and interpretability under non-uniform data conditions. This chapter is mainly based on the content in [70].

- Chapter 5 extend the framework presented in Chapter 2 leveraging kernel-based modeling. It integrates kernel methods with available physical models, thereby eliminating the need for predefined basis functions while preserving interpretability and regularization. Then, the kernel-based model is formulated in a state-space setting, combining it with unscented Kalman filtering and smoothing for latent state reconstruction. The proposed approach is validated on the benchmark dataset proposed in Chapter 2, highlighting improvements in predictive accuracy and simulation performance. This chapter draws from [71] and related forthcoming work.
- Chapter 6 summarizes the main contributions of the thesis and discusses their implications for the field of system identification.
- Finally, the appendices provide additional technical material supporting the main chapters. Appendix A details the multi-step optimization framework, including the use of automatic differentiation and computational complexity analysis. Appendix B presents a stability analysis of multi-step gradients and discusses the exploding gradient problem. Appendix C summarizes kernel approximation theory relevant to Chapter 5.

Chapter 2

The identification framework

Many real-world systems can be described by a combination of known physical models and unknown components that capture unmodeled dynamics or uncertainties. This chapter introduces the general modeling framework adopted in this thesis, retaining the knowledge derived from physical principles and allowing flexible and regularized correction terms to improve parameter and prediction accuracy.

The remainder of this chapter is organized into two subsections: the first introduces the general methodology and the problem setup, while the second presents two case studies to provide preliminary evidence of the framework's efficacy.

2.1 Problem setup

Let us consider a generic class of nonlinear, discrete-time dynamical systems, denoted by $\mathcal{S}$, described by a known model, derived, e.g., from physical principles, and an unknown term, accounting for unmodeled dynamics. The system of interest evolves according to the following equations:

$$\begin{aligned} \mathcal{S} \ : \ x_{k+1} &= f(x_k, u_k, \theta) + \Delta(x_k, u_k), \\ y_k &= h(x_k, u_k, \theta). \end{aligned} \tag{2.1}$$

Here, $x_k \in \mathbb{R}^{n_x}$ is the system state, $u_k \in \mathbb{R}^{n_u}$ the external input, and $y_k \in \mathbb{R}^{n_y}$ the observed output at time k. The system is characterized by a vector of unknown parameters $\theta \in \mathbb{R}^{n_\theta}$, which, together with the unknown initial state x_0, must be estimated. Functions $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_\theta} \to \mathbb{R}^{n_x}$ and $h : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_\theta} \to \mathbb{R}^{n_y}$ encode the portion of the model explicitly informed by physics, and describe the nominal state transition and output equations, respectively. In this context, the functional form of f and h is known *a priori*, as it directly derives from first principles and available knowledge of the system. These functions are assumed to be nonlinear, time-invariant, and continuously differentiable. In contrast, the function $\Delta : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ is completely

unknown and captures unmodeled dynamics, such as structural uncertainties, approximation errors, or neglected phenomena that cannot be captured by the known physical laws.

The identification goal is to determine an estimation model $\mathscr{M}$, depending on the estimated vector of parameters θ, that approximates the behavior of the true system $\mathcal{S}$. Importantly, this model should also be capable of compensating for the effects of unmodeled dynamics, i.e., the term Δ, whenever it is non-negligible. To determine $\mathscr{M}$, we rely on available sequences of T measured inputs and outputs, denoted respectively as $\tilde{\mathbf{u}}_{0:T-1} \doteq \{\tilde{u}_0, \dots, \tilde{u}_{T-1}\}$ and $\tilde{\mathbf{y}}_{0:T-1} \doteq \{\tilde{y}_0, \dots, \tilde{y}_{T-1}\}$. In particular, $\tilde{u}_k = u_k + v_k$, $\tilde{y}_k = y_k + w_k$, with $v_k \in \mathbb{R}^{n_u}$ and $w_k \in \mathbb{R}^{n_z}$ being the process and measurement noises, respectively. Note that, while $\tilde{u}_k$ is the known input, the actual input received by the system is u_k.

In the following, we introduce the general structure of the estimation model used to approximate the system dynamics, together with the associated cost function $\mathscr{C}_T$, which quantifies the cumulative prediction error over a finite time horizon T. This cost function will serve as the objective to be minimized in the corresponding optimization problem. To guide the identification toward models that are not only accurate but also physically meaningful and interpretable, additional terms are incorporated into $\mathscr{C}_T$. These include regularization terms that promote desired structural properties, such as sparsity in the black-box component used to approximate the unmodeled dynamics Δ, as well as penalty terms that encode known physical constraints, such as passivity, stability, or conservation laws. This formulation enables the integration of domain knowledge directly into the learning process, leading to more robust and physically consistent models.

2.1.1 Multi-step estimation model

To illustrate the modeling approach, we first consider a baseline scenario in which the unknown dynamics are absent, i.e., $\Delta(x, u) = 0$, for all $x \in \mathbb{R}^{n_x}$, $u \in \mathbb{R}^{n_u}$. This setting will be generalized in the following discussion. In this case, the system is fully described by the known functions f and h, and a natural choice for the estimation model is given by,

$$\begin{aligned} \bar{\mathscr{M}} \ : \ \hat{x}_{k+1} &= f(\hat{x}_k, u_k, \hat{\theta}), \\ \hat{y}_k &= h(\hat{x}_k, u_k, \hat{\theta}), \end{aligned} \tag{2.2}$$

where $\hat{x}_k \in \mathbb{R}^{n_x}$ and $\hat{y}_k \in \mathbb{R}^{n_y}$ denote the estimated state and output at time k, obtained using the identified parameters $\hat{\theta} \in \mathbb{R}^{n_\theta}$ and initial condition $\hat{x}_0 \in \mathbb{R}^{n_x}$.

Clearly, selecting this model when $\Delta(x, u)$ is not negligible would result in inaccurate predictions and potentially biased parameter estimates, as the estimation model would fail to capture the unmodelled dynamics. In particular, neglecting the term Δ during identification forces the parameter estimates to compensate for the missing dynamics, introducing bias in an attempt to reduce the prediction error. To address such a

case, within this framework, we define the estimation model $\mathscr{M}$ as the combination of the known physical dynamics in (2.1) and a black-box correction term δ, with the aim of approximating the unmodeled dynamics Δ. This leads to the following model:

$$\begin{aligned} \mathscr{M} \; : \; \hat{x}_{k+1} &= f(\hat{x}_k, u_k, \hat{\theta}) + \delta(\hat{x}_k, u_k, \omega), \\ \hat{y}_k &= h(\hat{x}_k, u_k, \hat{\theta}). \end{aligned} \tag{2.3}$$

Here, the term δ is parameterized by additional parameters ω, to be identified jointly with $\hat{\theta}$. The parameters ω may correspond, for instance, to the weights of a neural network [25], the coefficients of a linear regression model (e.g., ARX), or the parameters of a kernel-based regressor. Indeed, although in the initial setup we adopt a basis-function representation for δ, the proposed framework is general and can seamlessly accommodate more expressive classes of function approximators, such as kernel-based models (this extension will be treated in Chapter 5). Formally, let $\varphi \in \mathbb{R}^m$ be a vector of basis functions $\varphi(\hat{x}_k, u_k) = \left[\varphi_1(\hat{x}_k, u_k), \dots, \varphi_m(\hat{x}_k, u_k)\right]^\top$, with $\varphi_j : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}$. The ι-th element of δ is defined as

$$\delta_\iota = \sum_{j=1}^{m} \Omega_{\iota j} \varphi_j \Big(\sum_{i=1}^{n_x} W_{ij}^{(\iota)} \hat{x}_{i,k} + \sum_{i=1}^{n_u} W_{(n_x+i)j}^{(\iota)} u_{i,k} + B_{\iota j} \Big), \tag{2.4}$$

with $\iota \in \left[1, n_x\right]$. In this case, $\omega = \{\Omega, W, B\}$ consists of: (i) the basis functions weights $\Omega = [\Omega_{\iota j}] \in \mathbb{R}^{n_x, m}$; (ii) the input/state coefficients $W = [W^{(1)}, \dots, W^{(n_x)}] \in \mathbb{R}^{(n_x+n_u), n_x m}$ with $W^{(\iota)} = [W_{ij}^{(\iota)}] \in \mathbb{R}^{n_x+n_u, m}$; and (iii) the bias terms $B = [B_{\iota j}] \in \mathbb{R}^{n_x, m}$. Notice that, when available, domain knowledge and prior system understanding can guide the choice of the initial set of basis functions φ. In other cases, their choice can be carried out considering the many options in the literature (see, e.g., [72] for a discussion on basis functions and indications for their choice). Moreover, we note that elements of δ, Ω, W, and B can be selectively fixed or set to zero, allowing for arbitrary flexibility in the design of δ.

The aim of the black-box term δ is to augment the incomplete physical dynamics f by approximating Δ. Hence, δ must have only a complementary role, so that the better the available physical knowledge describes the physics of the system, the smaller δ will be. Moreover, besides minimizing the effect of the black-box part, we also aim to explain the model discrepancies with the simplest possible model. For this reason, a regularization term is introduced in the cost function, to ensure that the black-box component remains minimal when physical knowledge adequately describes the system, thus helping to maintain the interpretability and simplicity of the model.

In view of the formulation in (2.4), a suitable regularization is imposed on the matrix Ω in order to promote sparsity in the black-box correction term δ. Moreover, to extrapolate physical properties in the estimated model $\mathscr{M}$, as done, e.g., in many PINN-based approaches [27], we embed explicit physical penalty terms into the cost function. Thus, defining the prediction error sequence $\mathbf{e}_{0:T} = \{e_k\}_{k=0}^{T-1}$ with $e_k \doteq \hat{y}_k - y_k \in \mathbb{R}^{n_y}$,

$k \in [0, T-1]$, the multi-step cost function $\mathscr{C}_T$ is given by

$$\mathscr{C}_T(\theta, x_0, \omega; \mathbf{e}_{0:T-1}) \doteq \sum_{k=0}^{T-1} \Big(\mathscr{L}_k(\theta, x_0; e_k) + \gamma \|\Omega\|_1 + \lambda p(\hat{x}_k, \theta) + \nu q^2(\hat{x}_k, \theta) \Big). \quad (2.5)$$

Here, the local loss function $\mathscr{L}_k$, accounting for the prediction error e_k at time k, is assumed to be twice continuously differentiable, while the second term, i.e., $\|\Omega\|_1 \doteq \sum_{\iota=1}^{n_x} \|\Omega_\iota^\top\|_1$, being $\Omega_\iota^\top$ the ι-th row of Ω, is chosen to promote sparsity and ensure a simpler and interpretable representation of the unmodeled dynamics[1]. The weight $\gamma \in \mathbb{R}$ controls the regularization. Finally, the remaining terms p, q define additional physics-based cost. In particular, $\lambda, \nu \in \mathbb{R}$ are the penalty weights, and $p, q : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\theta} \to \mathbb{R}$ are twice continuously differentiable functions promoting $p(\hat{x}_k, \theta) \leq 0$, $q(\hat{x}_k, \theta) = 0$, $\forall k \in [0, T-1]$.

Based on the above formulation, we now define the resulting identification problem.

Problem 2.1 (Multi-step identification problem)

Given sequences $\tilde{\mathbf{u}}_{0:T-1}$ and $\tilde{\mathbf{y}}_{0:T-1}$, functions f and h in (2.1), and basis functions φ, estimate the optimal values for the physical parameters θ, initial state x_0, and black-box weights ω over the horizon T, such that the estimated model $\mathscr{M}$ (2.3) is the best physically-consistent approximation of $\mathscr{S}$ (2.1). Thus, given the multi-step cost $\mathscr{C}_T$ in (2.5), solve

$$(\theta^\star, x_0^\star, \omega^\star) \doteq \arg\min_{\theta, x_0, \omega} \ \mathscr{C}_T. \quad (2.6)$$

The formulation introduced above exhibits two distinctive features: a multi-step structure and an inherent physics-based nature.

On the one hand, the multi-step structure of Problem 2.1 has two immediate consequences. First, the evaluation of the cost function $\mathscr{C}_T(\cdot)$ requires simulating the candidate model (2.3) forward in time, implying that the loss at each time step depends on the decision variables both directly and indirectly through the propagated state estimates. Second, the computation of the gradient of $\mathscr{C}_T(\cdot)$ with respect to (θ, x_0, ω) is nontrivial, since the multi-step formulation introduces a recursive dependence of each state on all previous predictions, leading to high-order chains of Jacobian terms that must be handled efficiently to obtain accurate sensitivity information.

Appendix A provides a complete development of the proposed solution to these issues. In particular, Appendix A.2 derives a structured gradient recursion, motivated by automatic differentiation and Linear Parameter-Varying sensitivity propagation, that

[1] Being the ℓ_1-norm non-differentiable, gradient computation for optimization via first-order methods can possibly be enabled through, e.g., softplus smoothing [73] of the ℓ_1-norm.

enables the efficient computation of the multi-step gradients using first-order information. Subsequently, Appendix A.3 analyzes the computational complexity of the proposed recursion, demonstrating that the resulting first-order scheme remains tractable for realistically sized identification problems. Overall, Appendix A makes the identification problem (2.1) fully operational by transforming the conceptual formulation introduced here into a scalable algorithmic procedure.

Within this framework, the optimization problem (2.6) is therefore solved by means of first-order methods. Such algorithms, which under suitable conditions are guaranteed to converge to a (possibly local) solution, have recently gained popularity for their ability to handle large-scale and nonconvex problems efficiently. The proposed first-order approach, discussed in Appendix A, provides the computational backbone of the framework introduced in this chapter.

On the other hand, the proposed identification framework is inherently physics-based for two main reasons. First, the nominal dynamics f and h are directly derived from first-principles models or other established physical relationships, ensuring that the identified model retains physical interpretability. Second, physical information is explicitly embedded in the optimization through the penalty terms p and q appearing in the cost function (2.5). These terms enforce relevant physical properties, such as parameter bounds, stability, or conservation laws, thus constraining the optimization to remain consistent with known physics.

We now delve deeper into the role and construction of the functions p and q in (2.5), illustrating how domain-specific knowledge and physical principles can be embedded into the model through tailored penalty terms.

2.1.2 Physics-based penalty functions

To incorporate additional physical properties to the identified model $\mathscr{M}$, besides explicitly using the physical laws, we can follow the same philosophy adopted in many PINNs based approaches [27], i.e., to integrate physical constraints directly into the cost function, enabling a "*physics-guided learning*" that also leverages domain knowledge to inform the optimization process. In this way, the optimization is "steered" towards physically-consistent and meaningful solutions, thereby enhancing the robustness and reliability of the optimization outcome.

Specifically, this is achieved by inducing physical constraints through additional *penalty terms* introduced directly into the cost function $\mathscr{C}_T$, in order to keep the optimization problem unconstrained, as in (2.5). Thus, a range of physics-based constraints can be induced in the optimization problem by customizing the functions p and q. In the following, two simple examples are presented.

Convex constraints set in the parameters space

The first class of constraints we consider is rather natural, and consists of capturing physical limits on the model parameters θ. That is, we assume there exists known upper bound $\theta^{ub} = [\theta_i^{ub}]$, and lower bound $\theta^{lb} = [\theta_i^{lb}]$, $i \in [1, n_\theta]$, with $\theta_i^{ub} < \infty$, $\theta_i^{lb} > -\infty$, such that

$$\theta \in \Theta \doteq \left\{ \theta_i^{lb} \leq \hat{\theta}_i \leq \theta_i^{ub},\ i = [1, n_\theta] \right\}. \tag{2.7}$$

In such a case, an *exponential barrier function* can be used to define the physics-based penalty term as follows:

$$p(\hat{x}_k, \theta) \doteq \|e^{\alpha(\hat{\theta} - \theta^{ub})}\|_2^2 + \|e^{\alpha(\theta^{lb} - \hat{\theta})}\|_2^2, \tag{2.8}$$

where the positive scalar $\alpha \in \mathbb{R}$ represents a sharpness parameter, encouraging estimated parameter variables to remain within known intervals. Here, a special case is the parameter non-negativity constraint, where $\hat{\theta}_{i,k} \geq 0$ for all k, and (2.8) becomes $p(\hat{x}_k, \theta) \doteq \|e^{-\alpha\hat{\theta}_k}\|_2^2$. Alternatively, whenever Θ is a convex set, the identification algorithm can be modified by incorporating a projection step immediately after the parameters update, following the gradient computation.

Physical limits

In most cases, specific bounds for parameter values may not be known. However, physical limits of the state variable can be used to ensure that the model is physically consistent. Thus, similar bounding constraints are defined to enforce limits on the state variable being predicted. For instance, the constraint

$$x \in \mathcal{X} \doteq \left\{ x_i^{lb} \leq \hat{x}_i \leq x_i^{ub},\ i = [1, n_x] \right\}, \tag{2.9}$$

can be expressed with the following penalty term

$$p(\hat{x}_k, \theta) \doteq \|e^{\alpha(\hat{x}_k - x^{ub})}\|_2^2 + \|e^{\alpha(x^{lb} - \hat{x}_k)}\|_2^2,$$

encouraging the predicted state variables to stay within defined physical limits, or specific intervals where the trajectories are known to be stable.

Beyond promoting physical interpretability, these penalties also play an important role in the optimization. When the cost is evaluated over long horizons, the gradients associated with $\mathscr{C}_T(\cdot)$ may become poorly conditioned, a phenomenon analogous to exploding gradients observed in recurrent architectures. Appendix B analyzes this effect in detail, showing how constraining the predicted trajectories to remain within physically meaningful regions, for example, through barrier-like penalties on state and parameter ranges, helps regularize the multi-step gradient propagation and prevents instability in the optimization dynamics. This establishes a direct connection between the physically motivated constraints introduced in this section and the robustness of the optimization procedure described in Appendix A.

2.2 Case studies

To illustrate the practical effectiveness of the proposed modeling and identification framework, we now consider two representative case studies involving spacecraft and tank systems dynamics. In both case studies below, the parameters and the black-box weights are identified by solving Problem (2.1) using the first-order optimization strategy derived in Appendix A.

2.2.1 Spacecraft inertia identification

First, we present a study on the identification of the inertia tensor for a spacecraft, demonstrating the practical application of the combined physics-based and black-box framework introduced in this chapter.

Specifically, in this example, we consider a satellite whose rotational dynamics can be modeled in continuous time, exploiting the standard Euler equations, i.e.,

$$\mathcal{S} \; : \; \dot{x} = J_C^{-1} \left(u - x \times J_C x \right) + w. \tag{2.10}$$

Here, $x = [\omega_x \; \omega_y \; \omega_z]^\top$ is the satellite angular velocity expressed in the body frame, u represents the sum of external input, and $J_C = [J_{C,ij}] \in \mathbb{R}^{3\times3}$ is the inertia matrix to be identified. The term w represents noise.

According to Problem 2.1, the aim is to estimate the inertia matrix J_C and initial condition x_0 from T-step measured, input sequence $\tilde{\mathbf{u}}_{0:T-1}$ and the corresponding T collected observations $\tilde{\mathbf{y}}_{0:T-1}$, leading to a model $\mathscr{M}$ of the form (2.3), such that $\mathscr{M}$ is the best approximation of $\mathcal{S}$, given its underlying physical priors. Clearly, the setup in (2.3) can be recast from (2.10) having $y = x$, and letting θ denotes the elements of the inertial tensor. Hence, f can be immediately obtained by discretization of (2.10).

Inertia matrix physical properties

For the system under analysis, three physical constraints on the elements of the inertia matrix are accounted for, i.e., symmetry, positive definiteness, and the triangle inequality [74].

Symmetry. Symmetry can be enforced in two distinct ways. A first approach consists in imposing symmetry by design, by identifying the lower-triangular matrix and explicitly symmetrizing it in the model. Specifically, in this case, the identified inertia matrix is parametrized as

$$\hat{J}_C \doteq \begin{bmatrix} \hat{\theta}_1 & \hat{\theta}_2 & \hat{\theta}_4 \\ \hat{\theta}_2 & \hat{\theta}_3 & \hat{\theta}_5 \\ \hat{\theta}_4 & \hat{\theta}_5 & \hat{\theta}_6 \end{bmatrix}.$$

An alternative strategy, which will be briefly explored in this example, leverages recent advances in neural network theory related to benign overfitting [75], [76]. Although

overparametrization is not the main focus of this thesis, it provides a useful identification perspective for enforcing symmetry by introducing a redundant parameterization of the inertia matrix. Hence, in this case, the symmetry of the overparametrized inertia matrix, that is

$$\hat{J}_C \doteq \begin{bmatrix} \hat{\theta}_1 & \hat{\theta}_2 & \hat{\theta}_3 \\ \hat{\theta}_4 & \hat{\theta}_5 & \hat{\theta}_6 \\ \hat{\theta}_7 & \hat{\theta}_8 & \hat{\theta}_9 \end{bmatrix},$$

is imposed through the following set of equality constraints

$$\begin{aligned} q_1(\hat{x}_k, \theta) &\doteq \hat{\theta}_2 - \hat{\theta}_4 = 0, \\ q_2(\hat{x}_k, \theta) &\doteq \hat{\theta}_3 - \hat{\theta}_7 = 0, \\ q_3(\hat{x}_k, \theta) &\doteq \hat{\theta}_6 - \hat{\theta}_8 = 0. \end{aligned}$$

Remark 2.1 (On overparameterization)**.** *As discussed in work [60] and related works [75], [76], [77], redundant parameterizations can be exploited to encode structural symmetries (such as those of the inertia matrix) while maintaining physical consistency. In this view, overparameterization can be interpreted as a regularization mechanism that improves optimization landscape smoothness and convergence, without compromising the interpretability and reliability of the physical parameters. Interestingly, a similar effect may also emerge implicitly in the proposed multi-step identification framework. Indeed, the recursive propagation of the model over extended horizons introduces additional dependencies between the optimization variables across time, effectively enlarging the search space and providing multiple "views'' of the same parameters through their multi-step predictions. This implicit form of overparameterization may play a beneficial role in shaping the loss landscape, improving parameter identifiability and convergence behavior. This conjecture is currently under investigation.*

Positive semi-definiteness. The positive semi-definiteness of the inertia matrix is enforced by imposing the matrix to be diagonal dominant with real non-negative diagonal entries. Hence, the following additional inequality constraints are defined

$$\begin{aligned} p_1(\hat{x}_k, \theta) &\doteq |\hat{J}_{C,12}| + |\hat{J}_{C,13}| - |\hat{J}_{C,11}| \leq 0, \\ p_2(\hat{x}_k, \theta) &\doteq |\hat{J}_{C,21}| + |\hat{J}_{C,23}| - |\hat{J}_{C,22}| \leq 0, \\ p_3(\hat{x}_k, \theta) &\doteq |\hat{J}_{C,31}| + |\hat{J}_{C,32}| - |\hat{J}_{C,33}| \leq 0. \end{aligned} \tag{2.11}$$

Such non-differentiable constraints are effectively handled by projecting the parameters into the feasible parameter set defined by the inequalities whenever a violation occurs.

Triangle inequality. In addition to symmetry and positive semi-definiteness, we rely on the physical property of an inertia tensor defined by the triangle inequality. Indeed, as detailed in [74], if a symmetric, positive semi-definite real matrix does not satisfy the triangle inequality, it doesn't represent a physically possible distribution of

mass. Hence, triangle inequality constraints are imposed on the diagonal entries of the inertia matrix to prevent nonphysical predictions. These constraints are defined as follows

$$p_4(\hat{x}_k, \theta) \doteq \hat{J}_{C,11} - \hat{J}_{C,22} - \hat{J}_{C,33} \leq 0,$$
$$p_5(\hat{x}_k, \theta) \doteq \hat{J}_{C,22} - \hat{J}_{C,11} - \hat{J}_{C,33} \leq 0,$$
$$p_6(\hat{x}_k, \theta) \doteq \hat{J}_{C,33} - \hat{J}_{C,11} - \hat{J}_{C,22} \leq 0.$$

Numerical results

Having established the physical modeling framework and the corresponding constraint formulation, we now assess the effectiveness of the proposed approach through a dedicated simulation scenario. Specifically, we consider a 3U CubeSat with a mass of 4 kg, a (true) tensor of inertia

$$J_C = \begin{bmatrix} 0.04027 & 0.00312 & 0.000145 \\ 0.00312 & 0.04028 & 0.000971 \\ 0.000145 & 0.000971 & 0.00801 \end{bmatrix},$$

and initial condition $x_0 = [0, -0.0011, 0]^\top$. In this example, we rely on a strategy similar to the one proposed in [78] to generate a random input signal that persistently excites the system and simulates a tumbling motion useful for the identification of the inertia matrix. Hence, we consider $u \sim \mathcal{N}(10^{-5}, \sigma_u)$ with $\sigma_u = 10^{-7} \frac{\text{rad}}{\text{s}}$. The noise is assumed to be a Gaussian white noise $w \sim \mathcal{N}(0, \sigma_w)$ with standard deviation $\sigma_w = \left[5 \cdot 10^{-4}, 5 \cdot 10^{-4}, 3 \cdot 10^{-3}\right]^\top$ rad/s [2].

The following analysis involves the identification of the optimal values for the satellite inertia matrix using the proposed approach and a first-order gradient descent algorithm with an adaptive learning rate.

First, we consider the scenario where the physical model fully describes the true system dynamics, that is θ contains all the elements of the inertia matrix, and a model of the form (2.2). We generate a sequence of $T = 500$ data, integrating (2.10) with a sampling time $T_s = 0.1$s over a 50s simulation. Hence, we consider the forward Euler discretization of the same model as the physical model for the identification, i.e.,

$$x_{k+1} = x_k + T_s J_C^{-1} \left(M - x_k \times J_C x_k \right).$$

Here, we first investigate how different formulations of the symmetry penalty for the inertia matrix influence parameter identification. Following the approach outlined above, we first minimize the number of parameters to be estimated by defining $\hat{\theta}$ as the elements of the lower-triangular part of J_C, thereby enforcing symmetry directly in the

[2] The noise values are compatible with the case study selected (i.e., around 10% of the state values).

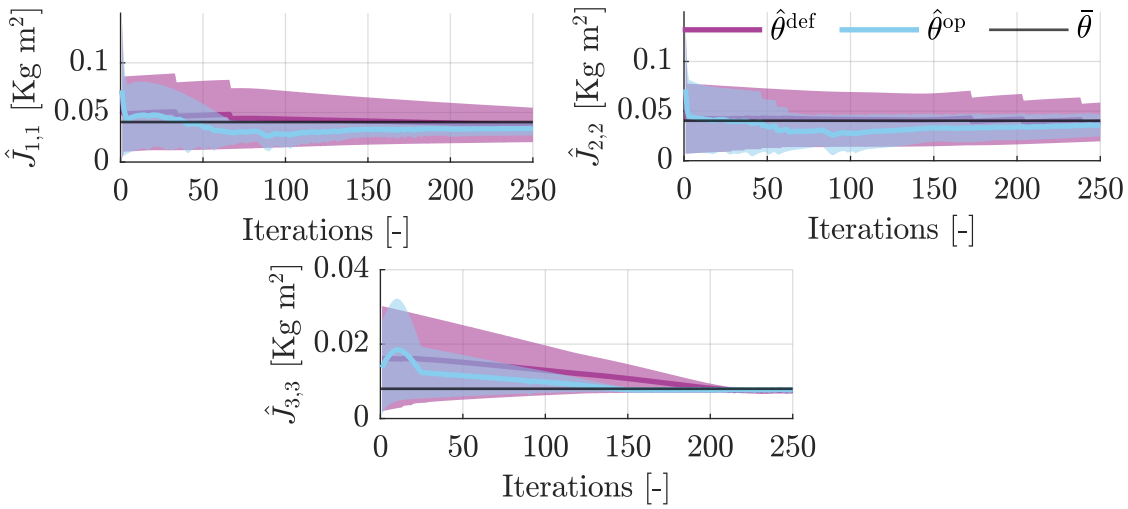


Figure 2.1: Comparison between estimated parameters $\hat{\theta}_i$ obtained relying on the complete physical model when symmetry is forced by definition (purple) and when overparametrization is exploited (blue).

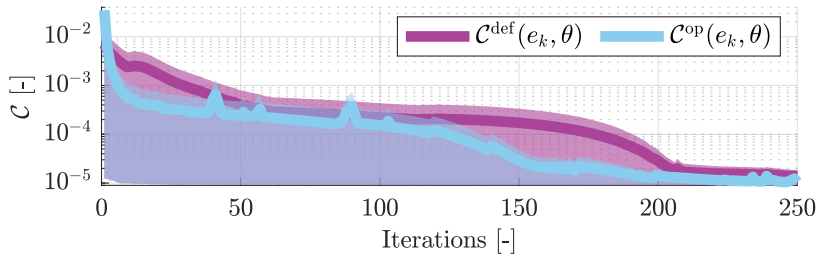


Figure 2.2: Evolution of the cost functions for the two approaches.

structure of the estimated inertia matrix. Then, we exploit the concept of "benign overfitting" increasing the number of parameters to identify (i.e., $\hat{\theta}$ contains all the elements of J_C) and forcing symmetry by additional equality constraints (2.11).

Fig. 2.1 shows the evolution of the estimated parameters with respect to the algorithm iterations for $N = 100$ different initial conditions of $\hat{\theta}$, randomly selected as $\hat{\theta}_0 \doteq \theta + \mathcal{N}(0, \theta)$.

It can be observed that, although both approaches provide a good mean estimation of the parameters, overparametrization leads to faster convergence and more accurate estimates. This reflects that the majority of the trajectories reach the vicinity of the true parameter value in a shorter time and with a lower estimation error. This behavior is confirmed by the evolution of the loss functions over 250 iterations, drawn in Fig. 2.2.

In the second scenario, we consider the same sequence of $T = 500$ data. However,

we model the satellite as a rigid, symmetric body, which simplifies the model by implicitly assuming the inertia matrix to be diagonal. This modeling assumption results in the following set of equations, discretized using the forward Euler method:

$$\begin{aligned}
\omega_{x,k+1} &= \omega_{x,k} + T_s \left(\frac{M_{x,k}}{J_{C,11}} - \frac{J_{C,33} - J_{C,22}}{J_{C,11}} \omega_{y,k} \omega_{z,k} \right), \\
\omega_{y,k+1} &= \omega_{y,k} + T_s \left(\frac{M_{y,k}}{J_{C,22}} - \frac{J_{C,11} - J_{C,33}}{J_{C,22}} \omega_{x,k} \omega_{z,k} \right), \\
\omega_{z,k+1} &= \omega_{z,k} + T_s \left(\frac{M_{z,k}}{J_{C,33}} - \frac{J_{C,22} - J_{C,11}}{J_{C,33}} \omega_{x,k} \omega_{y,k} \right).
\end{aligned} \tag{2.12}$$

This choice for $\mathscr{M}$ oversimplifies the model, considering only the dynamical terms that result from the diagonal entries of the inertia matrix and neglecting the dynamics arising from the off-diagonal terms. In this configuration, indeed, the unmodeled dynamics explicitly capture the coupling effects between the rotational axes that are omitted when assuming a diagonal inertia matrix. In this case, we compare the performance achievable with the purely physics-based estimation model, i.e., (2.12), and with the proposed modeling approach, where a black-box term of the form (2.4) is introduced to compensate for the unaccounted off-diagonal terms in the simplified model. In the latter case, (2.12) represents the term f in (2.3), while the selected vector of basis functions consists of sigmoid, softplus, hyperbolic tangent, and trigonometric functions. The regularization term in (2.5) is tuned with $\gamma = 0.1$, while $\lambda = \nu = 1$.

Fig. 2.3 shows the trajectories of the estimated parameters with respect to the algorithm iterations, calculated over $N = 100$ simulation for different parameter initial conditions. It is worth noting that, when the identification relies on a simplified and incomplete physical model, the parameters (θ^f) converge to incorrect values, compensating for the dynamic effects that are not considered. On the other hand, the black-box compensation is able to recover these effects, enabling physical parameters to converge to better estimates in a neighborhood of the true values with a higher convergence rate ($\theta^{f+\delta}$). Additionally, analyzing the parameters values for the initial iterations, it is possible to notice that the influence of the black-box terms appears to immediately steer the parameter values towards a more favorable region of the parameters space. This, in turn, allows reaching lower values of the cost function and better local minima, as shown in Fig. 2.4. In particular, it can be noticed that the mean value of the loss functions over all the trajectories is reduced by one order of magnitude when the black-box is exploited to compensate for missing terms in the physical model. However, it is important to remark that having access to the complete physical model results in faster convergence of all the parameters towards the true values (∼250 iterations compared to ∼600) with comparable mean squared errors, especially when involving overparameterization. This is shown in Table 2.1, where the average of the mean square error over $N = 100$ simulations is shown for all the proposed approaches.

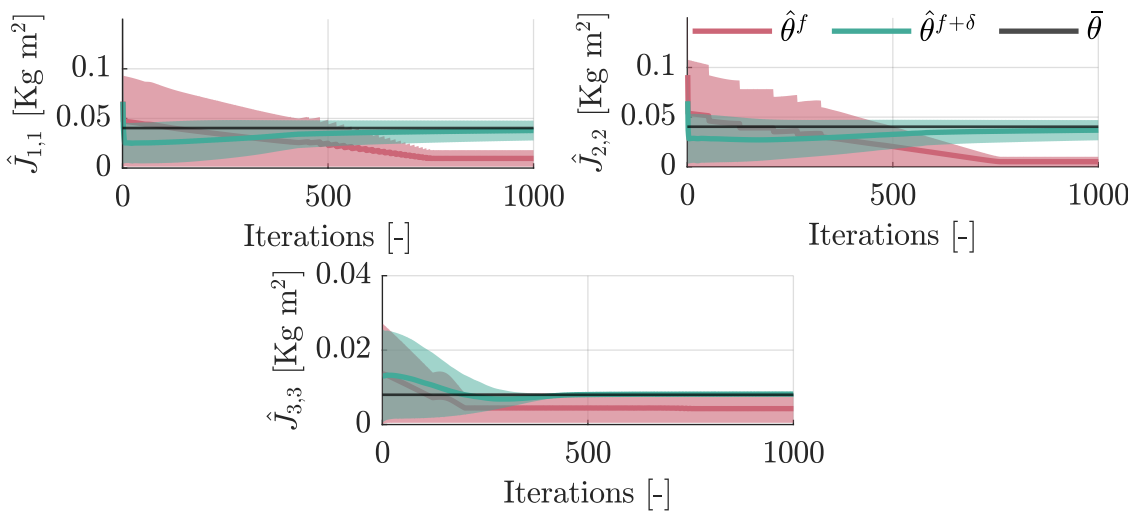


Figure 2.3: Comparison between estimated parameters $\hat{\theta}_i$ obtained relying on the given physics only and exploiting a black-box compensation.

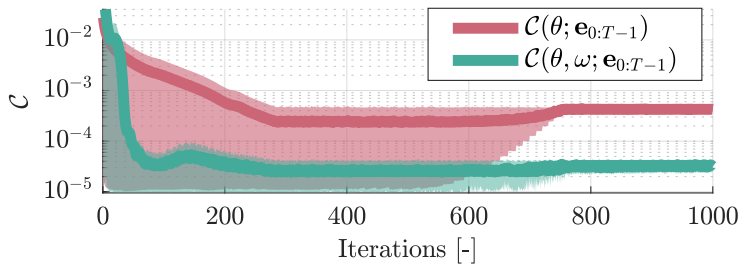


Figure 2.4: Evolution of the cost functions for the two approaches.

Table 2.1: Average MSE over 100 simulations.

Approach	e^{avg}_{MSE}
Full model	$1.66 \cdot 10^{-4}$
Full model + OP	$1.10 \cdot 10^{-5}$
Simple model	$6.00 \cdot 10^{-3}$
Simple model + BB	$3.70 \cdot 10^{-5}$

2.2.2 Cascade tank system identification benchmark

In this second case study, we evaluate the framework on the cascade tank system (CTS) benchmark, described in [79]. The CTS controls fluid levels using two connected tanks with free outlets and a pump. An input signal controls a water pump that transfers water from a reservoir to the upper tank. Then, the water flows through a small opening

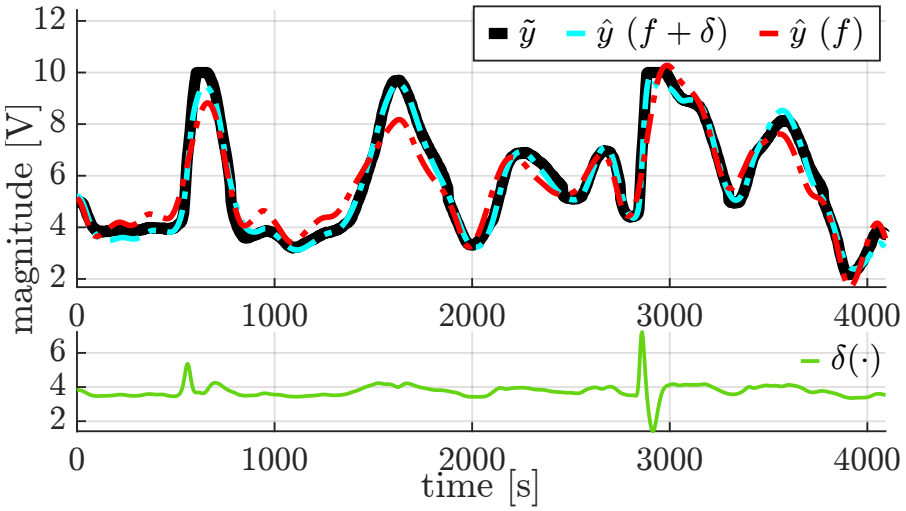


Figure 2.5: Validation data (simulation).

to the lower tank and back into the reservoir. Water overflow occurs when one tank is full. If the upper tank overflows, some water flows into the lower tank, while the rest leaves the system. The system follows a discrete-time, nonlinear model

$$\begin{aligned} x_{1,k+1} &= x_{1,k} + T_s(-k_1\sqrt{x_{1,k}} + k_4 u_k + v_{1,k}), \\ x_{2,k+1} &= x_{2,k} + T_s(k_2\sqrt{x_{1,k}} - k_3\sqrt{x_{2,k}} + v_{2,k}), \end{aligned} \tag{2.13}$$

with output $\tilde{y}_k = x_{2,k} + w_k$, input $u_k \in \mathbb{R}$, states $x_{1,k}, x_{2,k} \in \mathbb{R}$, and noise $v_{1,k}, v_{2,k}, w_k \in \mathbb{R}$. The unknown physical parameters $k = [k_1, k_2, k_3, k_4]$ and initial state values (equals for both training and validation) must be estimated from training data ($T = 1024$, $T_s = 4$s). Notice that some dynamics remain unmodeled since the model in (2.13) excludes overflow effects. Thus, the physical model is augmented with a black-box term comprising standard and most-used basis functions [72], such as sigmoid, softplus, hyperbolic, trigonometric, and polynomial functions. Then, a regularized cost function $\mathscr{C}_T$ ensures a minimal black box if the physical model (2.13) adequately describes the system. In this example, $\mathscr{L}_k = \frac{1}{T}\|e_k\|_2^2$ and $\gamma = 0.1$. Parameters are initialized as $\hat{k}_0 = [0.05, 0.05, 0.05, 0.05]$, $\hat{x}_0 = [y_0, y_0]^\top$. After the optimization, black-box parameters are set to 0 if $|\Omega_{ij}| \leq 10^{-4}$.

Fig. 2.5 compares measured and simulated outputs, showing how the black-box augmented model (cyan) effectively compensates for unmodeled dynamics, outperforming the physical-only model (red). This is confirmed by the Root Mean Square Error (RMSE), which improves from 0.603V (training) and 0.670V (validation) to 0.135V and 0.260V, respectively. Thus, to allow a fair comparison between different methods, as indicated in [79], accuracy is compared using the RMSE on the validation dataset, with

Table 2.2: Comparison of identification methods (RMSE on validation).

Method	RMSE	Method	RMSE
Svensson et al. [16]	0.45	PWARX [17]	0.35
Volt.FB [23]	0.39	SED-MPK [24]	0.48
INN [25]	0.41	PNLSS-I [80]	0.45
NLSS2 [80]	0.34	NOMAD [81]	0.37
Proposed	0.26		

predictions obtained through simulation. Table 2.2 compares our method with state-of-the-art identification approaches, achieving the lowest RMSE[3]. In particular, the proposed framework outperformed black-box kernel-based methods [23], [24] and black-box methods based on nonlinear state equations [80], as well as approaches based on NNs [25], linear-in-parameters models [16], [17], and methods based on derivative-free optimization techniques [81], demonstrating the effectiveness of the combination between off-white and black models. To further assess the identification accuracy, we report the optimal fitness percentage ($\text{fit}_{\%} = 100(1 - \|\hat{\boldsymbol{y}}_{0:T-1} - \tilde{\boldsymbol{y}}_{0:T-1}\|_2 / \|\hat{\boldsymbol{y}}_{0:T-1} - \bar{\boldsymbol{y}}\|_2)$, with $\bar{\boldsymbol{y}} \doteq \frac{1}{T}\sum_{k=0}^{T-1} \tilde{y}_k$): for the physics-only model, $\text{fit}_{\%}$ is 72.23% (training) and 68.16% (validation), while incorporating δ improves the accuracy to $\text{fit}_{\%} = 93.78\%$ (training) and 87.64% (validation). Identified parameters, whose true values are not reported in [79], are $\hat{k} = [0.076, 0.027, 0.042, 0.039]$, and $\hat{x}_0 = [3.52, 5.19]^\top$ V.

Discussion and concluding remarks

The framework introduced in this chapter establishes the foundations of this thesis. By combining known physical dynamics with a sparse data-driven term, it enables interpretable and flexible modeling of nonlinear systems while retaining consistency with prior knowledge. The multi-step formulation adopted throughout the identification process ensures long-horizon reliability and naturally integrates physical penalties and regularization terms within a unified optimization setting.

An intriguing aspect emerging from the present formulation, first discussed in Section 2.2.1, is that the multi-step propagation may itself act as an implicit form of over-parameterization. Indeed, by repeatedly embedding the model within the cost function over extended horizons, the optimization problem effectively enlarges the representation space, providing multiple temporal realizations of the same parameters. This mechanism can smooth the loss landscape and improve convergence behavior, similarly to explicit overparameterization strategies recently studied in learning theory [75], [76],

[3]To ensure metrics consistency, comparison methods provided at `https://github.com/MaartenSchoukens/nonlinear_benchmarks` are used.

[77]. Although this conjecture remains under investigation, it offers an interesting interpretation of the favorable convergence properties observed in practice.

From a computational viewpoint, the complete optimization methodology underlying the proposed framework is detailed in Appendix A, which reformulates the multi-step cost into a tractable first-order scheme through recursive gradient propagation and an analysis of its computational complexity. Complementarily, Appendix B examines the conditioning of the multi-step gradients and shows how the physically motivated penalty terms introduced in Section 2.1.2 enhance numerical stability during optimization. Together, these appendices provide the algorithmic and theoretical underpinnings that make the framework introduced in this chapter both practical and computationally efficient.

The next chapter builds upon this framework by establishing theoretical guarantees on parameter estimation accuracy and sparsity recovery, providing a rigorous foundation for the proposed formulation.

Chapter 3

Parameter theoretical guarantees and sparse modeling

Having introduced in the previous chapter a general framework for multi-step identification that integrates physics-based modeling with black-box correction terms, we now turn our attention to the theoretical aspects of this approach. In particular, this chapter focuses on two key aspects: (i) the derivation of bounds on the parameter estimation error, and (ii) the characterization of conditions under which sparse modeling of the unmodeled dynamics can be achieved. The first part of the chapter establishes guarantees on the accuracy of the identified physical parameters as a function of the noise level and the approximation quality of the black-box term. The second part investigates the recovery of sparse representations for the correction term, extending existing results from linear, single-step settings to the more general nonlinear, multi-step case considered here. Together, these contributions provide a rigorous foundation for the proposed identification strategy and justify the use of sparse black-box models in conjunction with physically interpretable dynamics, as demonstrated in the third part of the chapter. The final part illustrates the validity of the theoretical results through an academic example.

Before entering into the technical developments of this chapter, we introduce below the notation that will be consistently used throughout its sections. Given a vector $v \in \mathbb{R}^n$, $\|v\|_p$ denotes its ℓ_p-norm. The sequence $\mathbf{v}_{1:T} \doteq \{v_k\}_{k=1}^T$ represents the collection $\{v_1, \dots, v_T\}$, and $\|\mathbf{v}_{1:T}\|_p \doteq \|[v_1^\top, \dots, v_T^\top]^\top\|_p$ its stacked ℓ_p-norm. The support of a vector v is defined as $\mathrm{supp}(v) \doteq \{i \,:\, v_i \neq 0\}$, and its complement as $\overline{\mathrm{supp}}(v) \doteq \{i \,:\, v_i = 0\}$. Given a sequence $\mathbf{v}_{1:T}$ and a function $f(v_k)$, we write $f(\mathbf{v}_{1:T}) \doteq [f(v_1), \dots, f(v_T)]^\top$ to indicate the vector composed of the function evaluations over the sequence elements. For a matrix $A \in \mathbb{R}^{n,m}$, $\|A\|$ denotes its spectral norm, and $A^\dagger$ its Moore-Penrose pseudo-inverse. Finally, given $a, b \in \mathbb{N}$ with $a \leq b$, the interval $[a, b]$ denotes the integer set $\{a, a+1, \dots, b\}$.

3.1 Parametric error bound

We propose here key theoretical properties regarding the upper bound on the parametric identification error, i.e., the maximum distance between the solution $\theta^\star$ of the optimization problem (2.6) and the true parameters vector $\bar{\theta}$. In the following, we will highlight the relevant dependencies of $\mathscr{C}_T$, observing that

$$\mathscr{C}_T(\theta, x_0, \omega; \mathbf{e}_{0:T-1}) \equiv \mathscr{C}_T(\theta, x_0, \omega; \tilde{\mathbf{u}}_{0:T-1}, \tilde{\mathbf{y}}_{0:T-1}) \equiv \mathscr{C}_T(\theta, x_0, \omega; \boldsymbol{\eta})$$

with $\boldsymbol{\eta} = \{\boldsymbol{v}_{0:T-1}, \boldsymbol{w}_{0:T-1}\}$. Moreover, we assume for simplicity that x_0 is known and we focus on θ, noting that the general case extends straightforwardly. Clearly, due to nonlinearity and nonconvexity of the considered problem, we note that most results in this chapter are *local*, holding when the optimization is initialized sufficiently close to the optimal solution.

First, given that the solution of (2.6) may not be unique, we recall the concept of system *identifiability* [82], which determines the convergence behavior.

Definition 3.1 (System identifiability)**.** *A system $\mathcal{S}$ with parameters $\bar{\theta}$ and initial state $\bar{x}_0$ is locally identifiable if $\mathscr{C}_T$ has a strict local minimum at $\hat{\theta} = \bar{\theta}$, $\hat{x}_0 = \bar{x}_0$. If the minimum is global, the system is globally identifiable.*

Thus, we recall that a sufficient condition for local identifiability is that the Hessian matrix with respect to $\hat{\theta}$ and $\hat{x}_0$, is positive definite for all $\theta \in \Theta$, $x \in \mathscr{X}$, where Θ and $\mathscr{X}$ are suitable neighborhoods of $\bar{\theta}$, $\bar{x}_0$ [82].

To derive the parametric error bound, we rely on the following assumptions.

Assumption 3.1 (Basic assumptions)**.** *The following assumptions hold:*

i) Local identifiability: The system is locally identifiable according to Definition 3.1. Hence, the Hessian at $\theta^\star$ is positive definite for all noise realizations, i.e.,

$$H \doteq \frac{\partial^2 \mathscr{C}_T(\theta; \boldsymbol{\eta})}{\partial^2 \theta}\Big|_{\theta=\theta^\star} \succ 0, \quad \forall \boldsymbol{\eta}.$$

ii) Convergence to $\bar{\theta}$: When the noise is null and $\Delta = 0$, minimizing $\mathscr{C}_T$ recovers true parameters $\theta^\star = \bar{\theta}$.

Assumption 3.1.i) is rather standard in system identification, see, e.g., [82], [83], while Assumption 3.1.ii) holds when the optimization is initialized within the region of attraction of $\bar{\theta}$. Indeed, if $\Delta = 0$, i.e., no unmodeled dyanmics exist, the noise is null, i.e., $\boldsymbol{\eta} = \mathbf{0}$, and the identifiability assumption holds, then we have $\mathscr{C}_T(\bar{\theta}; \mathbf{0}) = 0$, making $\theta^\star = \bar{\theta}$ a local minimum.

Next, to proceed with the analysis, we formalize the boundedness of key quantities in the following lemma

Lemma 3.1 (Bounded function)

Let Assumption 3.1 hold. Assume $\boldsymbol{\eta} \in \mathcal{N}$ with $\mathcal{N}$ closed and bounded, and define $G(\boldsymbol{\eta}) \doteq \frac{\partial^2 \mathscr{C}_T(\theta;\boldsymbol{\eta})}{\partial \boldsymbol{\eta} \partial \theta}\big|_{\theta=\theta^\star}$.
Then $H^{-1}G(\boldsymbol{\eta})$ is bounded for all $\boldsymbol{\eta} \in \mathcal{N}$. Specifically, there exists constants $M_u, M_y < \infty$, such that

$$\max_{\boldsymbol{\eta}\in\mathcal{N}} \|[H^{-1}G(\boldsymbol{\eta})]_u\|_p < M_u, \quad \max_{\boldsymbol{\eta}\in\mathcal{N}} \|[H^{-1}G(\boldsymbol{\eta})]_y\|_p < M_y,$$

with $[H^{-1}G]_u$ and $[H^{-1}G]_y$ corresponding to the columns related to $\boldsymbol{v}_{0:T-1}$ and $\boldsymbol{w}_{0:T-1}$, respectively.

Proof. From Assumption *3.1.i)*, the Hessian H is positive definite $\forall \boldsymbol{\eta} \in \mathcal{N}$. Thus, it is invertible and its inverse is bounded. Moreover, $\mathscr{C}_T$ is twice continuously differentiable since it is a sum of twice differentiable functions with continuous derivatives $\mathscr{L}_k$. Hence, from Lipschitz continuity it follows that $G(\boldsymbol{\eta})$ is bounded. The lemma follows considering that $\mathcal{N}$ is closed and bounded. □

Thus, with the following theorem, we formalize an upper bound on the parametric identification error, highlighting its dependence on the noise and the black-box approximation quality.

Theorem 3.1 (Parametric error bound)

Define $\tilde{\boldsymbol{\Delta}} \doteq \tilde{\boldsymbol{\Delta}}_{0:T} = \{\widetilde{\Delta}_0, \dots, \widetilde{\Delta}_T\}$, with $\widetilde{\Delta}_k \doteq \Delta(x_k, u_k) - \delta(\hat{x}_k, u_k; \omega)$. Let Assumption 3.1 hold. Then, the parametric identification error is bounded as

$$\left\|\theta^\star - \bar{\theta}\right\|_p \leq M_u \left\|\boldsymbol{v}_{0:T}\right\|_p + M_y \left\|\boldsymbol{w}_{0:T}\right\|_p + M_\Delta \|\tilde{\boldsymbol{\Delta}}\|_p, \tag{3.1}$$

where M_Δ is a finite constant.

Proof. Consider system (2.1) with $\Delta = \delta = 0$ and true parameters $\bar{\theta} \in \mathbb{R}^{n_\theta}$. Since $\mathscr{C}_T$ is a function of the noise sequences $\boldsymbol{\eta}$, its minimizer $\theta^\star$ is also a function of $\boldsymbol{\eta}$, i.e., $\theta^\star \equiv \theta^\star(\boldsymbol{\eta})$. Therefore,

$$\theta^\star - \bar{\theta} = \theta^\star(\boldsymbol{\eta}) - \theta^\star(\mathbf{0}). \tag{3.2}$$

From the Mean Value Theorem, a $\breve{\boldsymbol{\eta}}$ exists such that

$$\theta^\star(\boldsymbol{\eta}) - \theta^\star(\mathbf{0}) = \frac{\partial \theta^\star(\breve{\boldsymbol{\eta}})}{\partial \boldsymbol{\eta}} \boldsymbol{\eta} = \frac{\partial \theta^\star(\breve{\boldsymbol{\eta}})}{\partial \boldsymbol{v}} \boldsymbol{v}_{0:T-1} + \frac{\partial \theta^\star(\breve{\boldsymbol{\eta}})}{\partial \boldsymbol{w}} \boldsymbol{w}_{0:T-1} \tag{3.3}$$

where $\frac{\partial\theta^\star(\breve{\boldsymbol{\eta}})}{\partial \boldsymbol{v}}$ and $\frac{\partial\theta^\star(\breve{\boldsymbol{\eta}})}{\partial \boldsymbol{w}}$ are the matrices splitting the contributions from process and measurement noise. Since $\theta^\star$ minimizes $\mathscr{C}_T$, it satisfies $\frac{\partial\mathscr{C}_T(\boldsymbol{\eta},\theta)}{\partial\theta} = 0$.

Hence, it follows from implicit differentiation that

$$\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\eta}}\frac{\partial\mathscr{C}_T(\boldsymbol{\eta},\theta)}{\partial\theta}\Big|_{\theta=\theta^\star} = \frac{\partial^2\mathscr{C}_T(\boldsymbol{\eta},\theta^\star)}{\partial\boldsymbol{\eta}\partial\theta} + \frac{\partial^2\mathscr{C}_T(\boldsymbol{\eta},\theta^\star)}{\partial^2\theta}\frac{\partial\theta^\star}{\partial\boldsymbol{\eta}} = G(\boldsymbol{\eta}) + H(\boldsymbol{\eta})\frac{\partial\theta^\star}{\partial\boldsymbol{\eta}} = 0.$$

Being the Hessian H invertible by assumption, then

$$\frac{\partial\theta^\star}{\partial\boldsymbol{\eta}} = -H^{-1}G(\boldsymbol{\eta}).$$

Thus, recalling the definitions of M_u and M_y in Lemma 3.1, it follows from (3.2) and (4.8) that

$$\left\|\theta^\star - \bar{\theta}\right\|_p \leq M_u \left\|\boldsymbol{v}_{0:T}\right\|_p + M_y \left\|\boldsymbol{w}_{0:T}\right\|_p. \tag{3.4}$$

Now, consider $\Delta \neq 0$. We observe that $\mathscr{C}_T$ is also a function of $\tilde{\boldsymbol{\Delta}}$ so that $\theta^\star \equiv \theta^\star(\boldsymbol{\eta}, \tilde{\boldsymbol{\Delta}})$. Thus, we have

$$\theta^\star - \bar{\theta} = \theta^\star(\boldsymbol{\eta}, \tilde{\boldsymbol{\Delta}}) - \theta^\star(\mathbf{0}, \tilde{\boldsymbol{\Delta}}) + \theta^\star(\mathbf{0}, \tilde{\boldsymbol{\Delta}}) - \theta^\star(\mathbf{0}, \mathbf{0}). \tag{3.5}$$

Applying the implicit function theorem [84] to $P = \frac{\partial\mathscr{C}_T(\mathbf{0},\widetilde{\Delta},\theta)}{\partial\theta}\Big|_{\theta=\theta^\star}$ it follows that $\theta^\star(\mathbf{0}, \tilde{\boldsymbol{\Delta}})$ is continuously differentiable with respect to $\tilde{\boldsymbol{\Delta}}$. Thus, by Lipschitz continuity, there exists a constant $M_\Delta < \infty$ such that

$$\|\theta^\star(\mathbf{0}, \tilde{\boldsymbol{\Delta}}) - \theta^\star(\mathbf{0}, \mathbf{0})\|_p \leq M_\Delta \|\tilde{\boldsymbol{\Delta}}\|_p. \tag{3.6}$$

Now, combining (3.4) for $\Delta \neq 0$ with (3.6), we have

$$\begin{aligned} &\|\theta^\star(\boldsymbol{\eta}, \tilde{\boldsymbol{\Delta}}) - \theta^\star(\mathbf{0}, \tilde{\boldsymbol{\Delta}})\|_p + \|\theta^\star(\mathbf{0}, \tilde{\boldsymbol{\Delta}}) - \theta^\star(\mathbf{0}, \mathbf{0})\|_p \\ &\quad\leq M_u \left\|\boldsymbol{v}_{0:T}\right\|_p + M_y \left\|\boldsymbol{w}_{0:T}\right\|_p + M_\Delta \|\tilde{\boldsymbol{\Delta}}\|_p. \end{aligned} \tag{3.7}$$

Finally, recalling the triangle inequality, (3.7) yields (3.1). □

Some interesting properties follow directly from Theorem 3.1. First, since the parametric identification error is inversely proportional to H^{-1}, a direct consequence of Lemma 3.1 and Theorem 3.1 is that a well-defined identification problem, characterized by a "large" invertible Hessian matrix H (see Definition 3.1), guarantees a "small" parametric identification error, thereby ensuring an accurate representation of the true system dynamics through the estimated parameters. Second, from Theorem 3.1 we can observe that as the black model $\delta(\cdot)$ more effectively compensates the effect of $\Delta(\cdot)$, that

is, as $\widetilde{\Delta}_k \to 0$, $\forall k$, the parametric error becomes more tightly bounded. This implies that an efficient compensation by $\delta(\cdot)$ leads to a reduction in the upper bounds of the parametric error, thus enhancing the accuracy of the physical parameters estimation. This aspect will be further examined in the subsequent discussion in Section 3.3. Moreover, while typically the unmodeled term $\Delta(x_k, u_k)$ is not known, some information about it may be available, e.g., an upper bound on its norm. Furthermore, it is reasonable to assume that the noises are bounded with known bounds. Thus, assuming that the following bounds are available, i.e.,

$$\begin{aligned} \|\tilde{\boldsymbol{\Delta}}\|_p &\leq \bar{\Delta} \\ \boldsymbol{\eta} \in \mathcal{N} \doteq \{\boldsymbol{\eta} : \left\|\boldsymbol{v}_{0:T-1}\right\|_p \leq \bar{v}, \left\|\boldsymbol{w}_{0:T}\right\|_p &\leq \bar{w}\}. \end{aligned} \tag{3.8}$$

we state the following corollary.

Corollary 3.1 (Bounded residuals)

Let Assumption 3.1 hold. Let $\boldsymbol{\eta} \in \mathcal{N}$ and $\|\tilde{\boldsymbol{\Delta}}\|_p \leq \bar{\Delta}$ (3.8). Then, the parametric identification error is bounded as

$$\left\|\theta^\star - \bar{\theta}\right\|_p \leq M_u \bar{v} + M_y \bar{w} + M_\Delta \bar{\Delta}.$$

Proof. The proof follows from easy argument directly considering Theorem (3.1) and bounds in (3.8). □

This section has established explicit bounds on the parametric identification error for the proposed multi-step, physics-informed framework, emphasizing its dependence on noise levels and, critically, on the approximation quality of the black-box component. These bounds provide formal guarantees on the reliability of the estimated physical parameters in the presence of unmodeled dynamics, thus motivating the sparse modeling analysis presented in the next section.

3.2 Maximum sparsity recovery

The parametric error bounds derived in the previous section have shown how the accuracy of the estimated physical parameters depends critically on the quality of the black-box approximation. This naturally motivates a closer examination of how such a component should be structured to maximize its effectiveness without compromising interpretability. In this light, this section focuses on the conditions under which the black-box term can be recovered in its sparsest possible form. The motivation behind pursuing a sparse representation of the unmodeled dynamics is threefold: first, to obtain a compact and parsimonious model that facilitates interpretability and physical

insight; second, to reduce model complexity and improve generalization by avoiding overfitting to noise or spurious patterns in the data; third, to ensure that the black-box component complements the available physical model without dominating it, by capturing only the residual dynamics that the physical description fails to explain.

To this end, we aim to characterize the structure of the sparsest black-box term that is capable of accurately compensating for the discrepancy between the physical model and the observed data. We begin our analysis by considering a simplified setting involving single-step prediction and a linear-in-parameters formulation of the black-box component, which allows for a tractable and insightful analysis. Then, the results are progressively extended to the more general and practically relevant multi-step prediction framework, where the model can be nonlinear in its parameters.

3.2.1 Single-step, linear-in-parameters setting

We consider a dynamical system described by the state-equation representation (2.1), with true parameters $\bar{\theta} \in \mathbb{R}^{n_\theta}$, a linear-in-parameters function f, full state observations, and, without loss of generality, $n_x = n_y = 1$. The system $\mathcal{S}$ takes the form

$$\begin{aligned} \mathcal{S} \; : \; x_{k+1} &= f(x_k, u_k, \bar{\theta}) + \Delta(x_k, u_k) = \xi^\top(x_k, u_k)\bar{\theta} + \Delta(x_k, u_k), \\ \tilde{y}_k &= x_k + \eta_k^z, \end{aligned} \tag{3.9}$$

with $\xi(x_k, u_k) : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_\theta}$ a vector of known (possibly nonlinear) physical terms defined as $\xi(x_k, u_k) = [\xi_1(x_k, u_k), \dots, \xi_{n_\theta}(x_k, u_k)]^\top$. A dictionary of basis $\varphi(x_k, u_k) = [\varphi_1(x_k, u_k), \dots, \varphi_m(x_k, u_k)]^\top$ and a set of noise-corrupted data $\mathscr{D} = \{\tilde{\mathbf{u}}_{0:T}, \tilde{\mathbf{y}}_{0:T}\}$, collected from (3.9), are available.

First, with the following lemma we formalize the measurement model underlying the relationship between input and output data.

Lemma 3.2 (Measurement model)

Consider a system (2.1) and a set of noise-corrupted data $\mathscr{D} = \{\tilde{\mathbf{u}}_{0:T}, \tilde{\mathbf{y}}_{0:T}\}$. The relationship between the measured input $\tilde{u}_k$ and the measured output $\tilde{y}_{k+1}$ is described by the following measurement model:

$$\tilde{y}_{k+1} = h(f(x_k, \tilde{u}_k, \bar{\theta}) + \Delta(x_k, \tilde{u}_k)) + d_k. \tag{3.10}$$

where d_k is an additive disturbance term accounting for both process noise v_k and measurement noise w_k.

Proof. Given (3.9), a sequence of collected inputs $\tilde{\mathbf{u}}_{0:T}$ applied to $\mathcal{S}$, and corresponding observations, $\tilde{\mathbf{y}}_{0:T}$, $\tilde{u}_k = u_k + v_k$, $\tilde{y}_k = y_k + w_k$, we have

$$\tilde{y}_{k+1} = h(f\left(x_k, u_k, \theta\right) + \Delta(x_k, u_k)) + w_{k+1}.$$

Let $\bar{h}\left(x_k, u_k, \theta\right) \doteq h(f\left(x_k, u_k, \theta\right) + \Delta(x_k, u_k))$. Substituting $u_k = \tilde{u}_k - v_k$, we obtain

$$\tilde{y}_{k+1} = \bar{h}\left(x_k, \tilde{u}_k - v_k, \theta\right) + w_{k+1}.$$

Thus, applying the mean value theorem, there exists a $\breve{u} \in [\tilde{u}_k - v_k, \tilde{u}_k]$ such that

$$\bar{h}\left(x_k, \tilde{u}_k - v_k, \theta\right) - \bar{h}\left(x_k, \tilde{u}_k, \theta\right) = \frac{\partial \bar{h}(\breve{u})}{\partial u_k} v_k,$$

so that, defining $d_k = \frac{\partial \bar{h}(\breve{u})}{\partial u_k} v_k + w_{k+1}$, that accounts for measurement and process noise, we can write the direct relationship between measured input and measured output as in (3.10), concluding the proof. □

Hence, according to Lemma 3.2, the measurement model for system (3.9) is given by

$$\tilde{y}_{k+1} = \xi^\top(x_k, \tilde{u}_k)\,\bar{\theta} + \Delta(x_k, \tilde{u}_k) + d_k.$$

Thus, we consider the following assumption on the disturbance term d_k.

Assumption 3.2 (Unknown but bounded noise)**.** *The noise sequence $d = [d_0, \dots, d_{T-1}]^\top$ is unknown but bounded, i.e., $\|d\|_2 \leq \mu$.*

We move then to the problem addressed in this section: finding a sparse linear combination of the basis functions in the dictionary that, combined with the prior physical model, is consistent with the measured data. First, we define the matrices $\Xi \in \mathbb{R}^{T,n_\theta}$ and $\Phi \in \mathbb{R}^{T,m}$ as follows:

$$\begin{aligned} \Xi &\doteq \begin{bmatrix} \xi_1(\tilde{x}_0, \tilde{u}_0) & \dots & \xi_{n_\theta}(\tilde{x}_0, \tilde{u}_0) \\ \vdots & \ddots & \vdots \\ \xi_1(\tilde{x}_{T-1}, \tilde{u}_{T-1}) & \dots & \xi_{n_\theta}(\tilde{x}_{T-1}, \tilde{u}_{T-1}) \end{bmatrix}, \\ \Phi &\doteq \begin{bmatrix} \varphi_1(\tilde{x}_0, \tilde{u}_0) & \dots & \varphi_m(\tilde{x}_0, \tilde{u}_0) \\ \vdots & \ddots & \vdots \\ \varphi_1(\tilde{x}_{T-1}, \tilde{u}_{T-1}) & \dots & \varphi_m(\tilde{x}_{T-1}, \tilde{u}_{T-1}) \end{bmatrix}, \end{aligned} \tag{3.11}$$

where $\tilde{x}_k \doteq \tilde{y}_k$, having initially considered full state measurements. Then, we state the definitions of feasible parameter set and maximally sparse coefficients.

Definition 3.2 (Feasible parameter set)**.** *Consider a dataset $\mathscr{D} = \{\tilde{\mathbf{u}}_{0:T}, \tilde{\mathbf{y}}_{0:T}\}$ satisfying Assumption 3.2, a sequence of states (predicted or measured) $\mathbf{x}_{0:T-1}$, and the noise bound μ. Let $\tilde{y} = [\tilde{y}_1, \dots, \tilde{y}_T]^\top$. We define the Feasible Parameter Set (FPS) as*

$$\text{FPS} \doteq \left\{\theta, \omega \,:\, \|\tilde{y} - f(\mathbf{x}_{0:T-1}, \tilde{\mathbf{u}}_{0,T-1}, \theta) - \Phi\omega\|_2 \leq \mu\right\}. \tag{3.12}$$

Moreover, for a given $\theta_0 \in \mathbb{R}^{n_\theta}$, the vector $\omega_0 \in \mathbb{R}^m$ is said to be feasible for θ_0 if $(\theta_0, \omega_0) \in$ FPS.

Definition 3.3 (Maximally sparse coefficients)**.** *Given $\mathscr{D} = \{\tilde{\mathbf{u}}_{0:T}, \tilde{\mathbf{y}}_{0:T}\}$ satisfying Assumption 3.2, $\mathbf{x}_{0:T-1}$, and the noise bound μ, a feasible coefficient vector is said* maximally sparse *if it is a solution of*

$$\begin{aligned} \bar{\omega} = \arg\min_{\theta,\omega} \ & \|\omega\|_0 \\ \text{s.t. } & \|\tilde{y} - f(\mathbf{x}_{0:T-1}, \tilde{\mathbf{u}}_{0,T-1}, \theta) - \Phi\omega\|_2 \leq \mu \end{aligned} \tag{3.13}$$

where $\|\cdot\|_0$ is the ℓ_0 quasi-norm.

We thus finally state the simplified sparsity problem.

Problem 3.1 (Simplified sparsity problem)

Consider a single-step, estimation model defined by the following state equation

$$\mathscr{M} \ : \quad \hat{x}_{k+1} = \sum_{i=1}^{n_\theta} \theta_i \xi_i(\tilde{x}_k, \tilde{u}_k) + \sum_{i=1}^{m} \omega_i \varphi_i(\tilde{x}_k, \tilde{u}_k),$$

expressed compactly as $\hat{x}_{k+1} = \hat{F}(\tilde{x}_k, \tilde{u}_k, \theta, \omega)$. The goal is to estimate $(\hat{\theta}, \hat{\omega})$ from the data set $\mathscr{D}$, such that:

i) $\hat{\omega}$ is sparse;

ii) $(\hat{\theta}, \hat{\omega})$ is consistent with the data, i.e., the single-step prediction error satisfies $\|\tilde{y} - \hat{F}(\tilde{\mathbf{x}}_{0:T-1}, \tilde{\mathbf{u}}_{0:T-1}, \hat{\theta}, \hat{\omega})\|_2 \leq \mu$, with μ is the noise bound from Assumption 3.2.

Note that no assumptions are made on the structural form of the black-box term Δ in (3.9). We only suppose that the chosen dictionary of basis function φ is sufficiently rich to allow an approximation of Δ compatible with the given noise level. Moreover, note that we do not explicitly require a statistical characterization of the noise. Instead, we only assume that an upper bound on its norm is given.

We now move on how Problem 3.1 can be addressed. Solving the sparse identification problem via (3.13) is intractable due to the nonconvexity of the ℓ_0 quasi-norm, which makes it NP-hard. Instead, we consider the following convex relaxation

$$\begin{aligned} (\hat{\theta}, \hat{\omega}) = \arg\min_{\theta,\omega} \ & \|\omega\|_1 \\ \text{s.t.} & \|\tilde{y} - \hat{F}(\tilde{\mathbf{x}}_{0:T-1}, \tilde{\mathbf{u}}_{0:T-1}, \theta, \omega)\|_2 \leq \mu. \end{aligned} \tag{3.14}$$

This ℓ_1-regularized problem encourages sparsity in $\hat{\omega}$, addressing condition (i) in Problem 3.1. However, it does not necessarily yield a maximally sparse solution in the sense of Definition 3.3, i.e., the sparsest set of basis functions that still ensures consistency with the data. In the following, we analyze the conditions under which a sparse solution of (3.14) is also maximally sparse.

3.2.2 Theoretical sparsity analysis

In [85], [86], the authors established conditions under which a coefficient vector, representing the solution to a general sparsity problem, is maximally sparse. While their focus is on sparsifying all decision variables, in this thesis we extend these results considering a setting where only the black-box coefficients ω (a subset of the decision variables (θ, ω)), require sparsification. Additionally, we generalize these results from single-step, linear systems to multi-step, nonlinear cases (see Subsection 3.2.3). First, in the following, we introduce some key concepts.

Definition 3.4 (Preliminary notations)**.** *For each integer $n \in \mathbb{R}$ and matrix $Q \in \mathbb{R}^{n_1,n_2}$ we denote*

$$\sigma_n^2(Q) \doteq \inf_{\|x\|_0 \leq n} \frac{\|Qx\|_2^2}{\|x\|_2^2}, \quad \|x\|_{(Q,n)} \doteq \sqrt{\sum_{i \in \mathcal{I}_n(x)} \left(x^\top q_i\right)^2},$$

where $\mathcal{I}_n(x)$ indexes the n largest inner products $|x^\top q_i|$, with q_i the i-th column of Q.

Definition 3.5 (Prediction errors)**.** *Given observation $y \in \mathbb{R}^T$ and matrices $P \in \mathbb{R}^{T,n_\theta}$ and $Q \in \mathbb{R}^{T,m}$ such that $y = P\theta + Q\omega$, the prediction error is defined as*

$$e_{P,Q}(y, \theta, \omega) \doteq y - P\theta - Q\omega.$$

Moreover, given $\theta^\star_{P,Q}(y, \omega) \doteq \arg\min_{\theta \in \mathbb{R}^{n_\theta}} \|e_{P,Q}(y, \theta, \omega)\|_2$, the optimal compensation error is defined as

$$e^\star_{P,Q}(y, \omega) \doteq e_{P,Q}(y, \theta^\star_{P,Q}(y, \omega), \omega).$$

For given y, P, Q, the prediction error is the discrepancy between measurements and predictions when employing black-box augmentation. The corresponding optimal compensation error, $e^\star_{P,Q}(y, \omega)$, represents the remaining prediction error after substituting the optimal estimate $\theta^\star_{P,Q}(y, \omega)$, explicitly depending on ω, into the model. Given Definition 3.5, the following Lemma holds.

Lemma 3.3 (Optimal compensation error)

Let $\Upsilon(P) = (I_T - PP^\dagger)$. Considering Definition 3.5, it holds that

$$\theta^\star_{P,Q}(y, \omega) = P^\dagger(y - Q\omega).$$

Moreover, $e^\star_{P,Q}(y, \omega) = \Upsilon(P)(y - Q\omega)$.

Proof. Solving $\min_{\theta \in \mathbb{R}^{n_\theta}} \|e_{P,Q}(y, \theta, \omega)\|_2 = \|(y - Q\omega) - P\theta\|_2$, yields the least squares solution, i.e.,

$$\theta^\star_{P,Q}(y, \omega) = (P^\top P)^{-1} P^\top (y - Q\omega) = P^\dagger(y - Q\omega).$$

Thus, $e^{\star}_{P,Q}(y,\omega)$ simplifies to

$$\begin{aligned} e_{P,Q}(y,\theta^{\star}_{P,Q}(y,\omega),\omega) &= y - PP^{\dagger}(y-Q\omega) - Q\omega \\ &= (I_T - PP^{\dagger})y - (I_T - PP^{\dagger})Q\omega, \end{aligned}$$

concluding the proof. □

Thus, to proceed with the analysis of the sparsity recovery conditions, we introduce the following additional assumption, which ensures problem feasibility.

Assumption 3.3 (Problem feasibility)**.** *The following assumptions hold:*

i) FPS $\neq \emptyset$*, i.e., the Feasible Parameters Set is not empty.*

ii) For given y*,* $\boldsymbol{P}$*,* $\boldsymbol{Q}$*, a feasible* ω *exists for* $\theta^{\star}_{P,Q}(y,\omega)$ *according to Definition 3.2.*

Building on the setup defined up to this point, the following theorems provide conditions under which the solution of (3.14) shares the same support as $\bar{\omega}$, ensuring maximal sparsity. First, using results from [85, Theorem 1, Corollary 1], we formally establish when two vectors have equivalent supports.

Theorem 3.2 (Equivalent support conditions)

Given $\mathscr{D} = \{\tilde{\mathbf{u}}_{0:T}, \tilde{\mathbf{y}}_{0:T}\}$, and Ξ, Φ (3.11), let Assumptions 3.1, 3.2, 3.3 hold. Let $\hat{\omega} \in \mathbb{R}^m$ be the solution of (3.14), with $M \doteq \|\hat{\omega}\|_0$ the cardinality of its support, and $\Upsilon_0 = \Upsilon(\Xi)$. Consider any other representation ω' and the noise-corrupted observation vector $\tilde{y} = [\tilde{y}_1, \dots, \tilde{y}_T]^{\top}$ from (3.9). If the conditions

$$\begin{aligned} \|e^{\star}_{\Xi,\Phi}(\tilde{y},\omega')\|_2 &\leq \|e^{\star}_{\Xi,\Phi}(\tilde{y},\hat{\omega})\|_2, \\ \|\omega'\|_0 &\leq M, \end{aligned}$$

hold, then,

$$\|\omega' - \hat{\omega}\|_{\infty} \leq \frac{\|e^{\star}_{\Xi,\Phi}(\tilde{y},\hat{\omega})\|_{(\Upsilon_0,1)} + \|e^{\star}_{\Xi,\Phi}(\tilde{y},\hat{\omega})\|_{(\Upsilon_0,2M)}}{\sigma^2_{2M}(\Upsilon_0)}.$$

Moreover, if

$$\|e^{\star}_{\Xi,\Phi}(\tilde{y},\hat{\omega})\|_{(\Upsilon_0,1)} + \|e^{\star}_{\Xi,\Phi}(\tilde{y},\hat{\omega})\|_{(\Upsilon_0,2M)} < \sigma^2_{2M}(\Upsilon_0)\eta(\hat{\omega}),$$

with $\eta(\omega) \doteq \min_{i\in\mathrm{supp}(\omega)} |\omega_i|$, then for all $i \in [1,m]$, $\hat{\omega}$ and ω' have the same support and sign, i.e., $\mathrm{supp}(\hat{\omega}) = \mathrm{supp}(\omega')$ and $\mathrm{sign}(\hat{\omega}_i) = \mathrm{sign}(\omega'_i)$.

Proof. The optimization problem (3.14) is a *feasibility problem* in θ. Therefore, we equally reformulate it by selecting ω for which a θ exists that satisfies the constraint, i.e.,

$$\begin{aligned}\hat{\omega} = \arg\min_{\omega} \|\omega\|_1,\\ \text{s.t. } \omega \in \left\{\omega : \exists \theta \text{ s.t. } \|\tilde{y} - \Xi\theta - \Phi\omega\|_2 \leq \mu\right\}.\end{aligned}$$

Moreover, since at least one $\theta \in \mathbb{R}^{n_\theta}$ satisfies the constraint, we further refine the problem by seeking the ω for which a suitable $\theta \in \mathbb{R}^{n_\theta}$ minimizes the error, i.e.,

$$\hat{\omega} = \arg\min_{\omega} \|\omega\|_1 \tag{3.15a}$$

$$\text{s.t. } \omega \in \{\omega : \min_{\theta \in \mathbb{R}^{n_\theta}} \|\tilde{y} - \Xi\theta - \Phi\omega\|_2 \leq \mu\}. \tag{3.15b}$$

The optimal θ minimizing $\|\tilde{y} - \Xi\theta - \Phi\omega\|_2$ is given the least squares optimal solution, i.e. $\theta^\star(\omega) = \Xi^\dagger(\tilde{y} - \Phi\omega)$, that once substituted in (3.15b) gives

$$\begin{aligned} g(\omega) &= \|\tilde{y} - \Xi\Xi^\dagger(\tilde{y} - \Phi\omega) - \Phi\omega\|_2 \\ &= \|\tilde{y} - \Xi\Xi^\dagger\tilde{y} + \Xi\Xi^\dagger\Phi\omega - \Phi\omega\|_2 \\ &= \|(I_T - \Xi\Xi^\dagger)\tilde{y} - (I_T - \Xi\Xi^\dagger)\Phi\omega\|_2 \\ &= \|\Upsilon_0(\tilde{y} - \Phi\omega)\|_2 = \|e^\star_{\Xi,\Phi}(\tilde{y}, \omega)\|_2, \end{aligned}$$

recalling $\Upsilon_0 = (I_T - \Xi\Xi^\dagger)$ (see Lemma 3.3). Thus, (3.15) simplifies to

$$\begin{aligned}\hat{\omega} = \arg\min_{\omega} \|\omega\|_1\\ \text{s.t. } \|\Upsilon_0(\tilde{y} - \Phi\omega)\|_2 \leq \mu.\end{aligned}$$

By applying [85, Theorem 1, Corollary 1], with $\Upsilon_0\hat{y} = \Upsilon_0\Phi\hat{\omega} + e^\star_{\Xi,\Phi}(\tilde{y}, \hat{\omega})$, the result follows. □

Theorem 3.2 provides verifiable conditions to determine whether two candidate solutions share the same sparsity pattern, i.e., identical support and the same sign pattern. Moreover, beyond this specific setting, the conditions of Theorem 3.2 retain a broader applicability, as highlighted in the following remark.

Remark 3.1 (Generality of the sparsity conditions)**.** *While Theorem 3.2 specifically applies results from [85] to* (3.14)*, its conditions can be verified on any estimate $\hat{\omega} \in \mathbb{R}^m$ satisfying $\Upsilon_0\tilde{y} = \Upsilon_0\Phi\hat{\omega} + e^\star_{\Xi,\Phi}(\tilde{y}, \hat{\omega})$.*

To proceed, let us define the vector ω^{υ} as the solution of

$$\omega^{\upsilon} \doteq \arg\min_{\omega} \|\omega\|_1 \tag{3.16a}$$

$$\text{s.t.} \quad \omega_i \geq \text{sign}(\hat{\omega}_i)\eta(\hat{\omega}), \quad \forall i \in \text{supp}(\hat{\omega}) \tag{3.16b}$$

$$|\omega_i| < \eta(\hat{\omega}), \quad \forall i \in \overline{\text{supp}}(\hat{\omega}) \tag{3.16c}$$

$$\|\Upsilon_0 \tilde{y} - \Upsilon_0 \Phi \omega\|_2 \leq \mu, \tag{3.16d}$$

where $\eta(\omega) \doteq \min_{i \in \text{supp}(\omega)} |\omega_i|$. Building on the results from [86, Theorem 1], with the following theorem we define conditions ensuring that a vector is maximally sparse.

Theorem 3.3 (maximum sparsity recovery)

Given $\mathscr{D} = \{\tilde{\mathbf{u}}_{0:T}, \tilde{\mathbf{y}}_{0:T}\}$, and Ξ, Φ (3.11), let Assumptions 3.1, 3.2, 3.3 hold. Let $\bar{\omega}$ and ω^{υ} be the solutions of (3.13) and (3.16), respectively. Let $\hat{\omega}$ be the parameter vector obtained from (3.14), $M \doteq \|\hat{\omega}\|_0$ the cardinality of its support, and $\Upsilon_0 = \Upsilon(\Xi)$. Define the noise-corrupted observation vector from (3.9) as $\tilde{y} = [\tilde{y}_1, \dots, \tilde{y}_T]^{\top}$, and let $\kappa_e \doteq \|\hat{\omega}\|_0 - \|\bar{\omega}\|_0$. Define the set

$$\lambda \doteq \left\{ i \,:\, |\omega_i^{\upsilon}| > \frac{\|e^{\star}_{\Xi,\Phi}(\tilde{y}, \omega^{\upsilon})\|_{(\Upsilon_0,1)} + \|e^{\star}_{\Xi,\Phi}(\tilde{y}, \omega^{\upsilon})\|_{(\Upsilon_0,2M)}}{\sigma^2_{2M}(\Upsilon_0)} \right\}.$$

Assume that the constraint (3.16d) is active, i.e., $\|\Upsilon_0 \tilde{y} - \Upsilon_0 \Phi \omega^{\upsilon}\|_2 = \mu$. Then,

$$\kappa_e \leq \bar{\kappa}_e \doteq \|\hat{\omega}\|_0 - card(\lambda). \tag{3.17}$$

Moreover, if

$$\frac{\|e^{\star}_{\Xi,\Phi}(\tilde{y}, \omega^{\upsilon})\|_{(\Upsilon_0,1)} + \|e^{\star}_{\Xi,\Phi}(\tilde{y}, \omega^{\upsilon})\|_{(\Upsilon_0,2M)}}{\sigma^2_{2M}(\Upsilon_0)} < \eta(\hat{\omega}), \tag{3.18}$$

then $\hat{\omega}$ is maximally sparse and $\text{supp}(\hat{\omega}) = \text{supp}(\bar{\omega})$.

Proof. The optimization problem (3.13) with respect to ω can be written as

$$\bar{\omega} = \arg\min_{\omega \in \mathbb{R}^m} \|\omega\|_0$$
$$\text{s.t.} \quad \|\Upsilon_0 \tilde{y} - \Upsilon_0 \Phi \omega\|_2 \leq \mu.$$

By definition, $\bar{\omega}$ is the sparsest vector satisfying the constraint $\|\Upsilon_0 \tilde{y} - \Upsilon_0 \Phi \omega\|_2 \leq \mu$. Thus, consider ω^{υ}, solution of (3.16) [86]. Since constraint (3.16d) is active, we have

$$\|e^{\star}_{\Xi,\Phi}(\tilde{y}, \bar{\omega})\|_2 \leq \|e^{\star}_{\Xi,\Phi}(\tilde{y}, \omega^{\upsilon})\|_2 = \mu.$$

Moreover, $\|\bar{\omega}\|_0 \leq \|\omega^v\|_0$, since $\bar{\omega}$ is maximally sparse. Applying Theorem 3.2,

$$\|\bar{\omega} - \omega^v\|_\infty \leq \frac{\|e^\star(\tilde{y}, \omega^v)\|_{(\Upsilon_0,1)} + |e^\star(\tilde{y}, \omega^v)|_{(\Upsilon_0,2M)}}{\sigma^2_{2M}(\Upsilon_0)}$$

holds, implying that, for all $i \in [1, m]$,

$$|\bar{\omega}_i - \omega^v_i| \leq \frac{\|e^\star(\tilde{y}, \omega^v)\|_{(\Upsilon_0,1)} + \|e^\star(\tilde{y}, \omega^v)\|_{(\Upsilon_0,2M)}}{\sigma^2_{2M}(\Upsilon_0)}.$$

Hence, if

$$|\omega^v_i| > \frac{\|e^\star(\tilde{y}, \omega^v)\|_{(\Upsilon_0,1)} + \|e^\star(\tilde{y}, \omega^v)\|_{(\Upsilon_0,2M)}}{\sigma^2_{2M}(\Upsilon_0)},$$

then $\bar{\omega} \neq 0$ and, consequently

$$\lambda \doteq \{i \,:\, |\omega^v_i| > \frac{\|e^\star(\tilde{y}, \omega^v)\|_{(\Upsilon_0,1)} + \|e^\star(\tilde{y}, \omega^v)\|_{(\Upsilon_0,2M)}}{\sigma^2_{2M}(\Upsilon_0)}\} \subseteq \text{supp}(\bar{\omega}).$$

If follows that $\|\bar{\omega}\|_0 \geq \text{card}(\lambda)$, which yields (3.17). Now, the constraints (3.16b) and (3.16c) imply that $\text{supp}(\hat{\omega}) = \{i \,:\, |\omega^v_i| \geq \eta(\hat{\omega})\}$. Moreover, if condition (3.18) holds, then

$$\{i \,:\, |\omega^v_i| \geq \eta(\hat{\omega})\} \subseteq \lambda \subseteq \text{supp}(\bar{\omega}).$$

It follows that $\text{supp}(\hat{\omega}) \subseteq \text{supp}(\bar{\omega})$. Since $\bar{\omega}$ is the sparsest vector satisfying $\|\Upsilon_0\tilde{y} - \Upsilon_0\Phi\omega\|_2 \leq \mu$ relation $\text{supp}(\hat{\omega}) = \text{supp}(\bar{\omega})$ follows, concluding the proof. □

3.2.3 Multi-step, nonlinear-in-parameters extension

We now consider the more general case where the system evolves according to (2.1) with true parameters $\bar{\theta}$, $n_x = n_z = 1$, and no assumptions on the forms of f and h. A dictionary of basis function $\varphi \in \mathbb{R}^m$ and a noise-corrupted dataset $\mathscr{D} = \{\tilde{\mathbf{u}}_{0:T}, \tilde{\mathbf{y}}_{0:T}\}$ are available. The measurement model, as detailed by Lemma 3.2, is given by

$$\tilde{y}_{k+1} = h(f(x_k, \tilde{u}_k, \bar{\theta}) + \Delta(x_k, \tilde{u}_k)) + d_k.$$

First, the following problem is stated, while the nonlinearity of the considered setting is highlighted in the subsequent remark.

Problem 3.2 (Multi-step sparsity problem)

Consider a multi-step estimation model defined as

$$\mathscr{M} : \quad \hat{x}_{k+1} = f(\hat{x}_k, \tilde{u}_k, \theta) + \sum_{i=1}^{m} \omega_i \varphi_i(\hat{x}_k, \tilde{u}_k), \tag{3.19}$$

expressed compactly as $\hat{x}_{k+1} = \hat{F}(\hat{x}_k, \tilde{u}_k, \theta, \omega)$ with output $\hat{y}_k = h(\hat{x}_k)$. The goal is to estimate $(\hat{\theta}, \hat{\omega})$ such that:

i) $\hat{\omega}$ is sparse;

ii) $(\hat{\theta}, \hat{\omega})$ is consistent with the dataset, i.e., $\|\tilde{y} - \hat{y}\|_2 \leq \mu$, with $\hat{y} = [\hat{y}_1, \dots, \hat{y}_T]^\top$.

Remark 3.2 (Multi-step nonlinearity)**.** *In the multi-step setting ($T > 1$), iterating the system over time introduces strong nonlinearity in the parameters, even if $\hat{F}$ is linear in θ and ω. Specifically, the k-th output prediction follows $\hat{y}_k = h(\hat{F}^k(\hat{x}_0, \tilde{u}_1, \dots, \tilde{u}_{k-1}, \theta, \omega))$, where $\hat{F}^k$ represents the k-th iteration of* (3.19).

To solve Problem 3.2, we consider the optimization problem (3.13), using the predicted sequence $\hat{\mathbf{x}}_{0:T-1}$ instead of state measurements, which are not available in this setting. As in the single-step case, we solve the following convex relaxation

$$(\hat{\theta}, \hat{\omega}) = \quad \arg\min_{\theta,\omega} \|\omega\|_1 \quad \text{s.t. } \|\tilde{y} - \hat{y}\|_2 \leq \mu. \tag{3.20}$$

In the following, we extend Theorems 3.2 and 3.3 to the multi-step case, under Assumptions 3.2 and 3.3. From a conceptual point of view, this is nontrivial, as the parameters appear inside highly nonlinear functions.

Theorem 3.4 (local equivalent support conditions)

Given $\mathscr{D} = \{\tilde{\mathbf{u}}_{0:T}, \tilde{\mathbf{y}}_{0:T}\}$, let Assumptions 3.1, 3.2, 3.3.i) hold. Let $\hat{\omega} \in \mathbb{R}^m$ be a local solution of (3.20) with $M \doteq \|\hat{\omega}\|_0$. Consider any other representation ω' in a local neighborhood of $\hat{\omega}$. Define the Jacobian matrices

$$\mathscr{J}_\theta \doteq \frac{\partial h(\hat{F}(\hat{\mathbf{x}}_{0:T-1}, \tilde{\mathbf{u}}_{0:T-1}, \hat{\theta}, \hat{\omega}))}{\partial \theta} \in \mathbb{R}^{T, n_\theta}, \quad \mathscr{J}_\omega \doteq \frac{\partial h(\hat{F}(\hat{\mathbf{x}}_{0:T-1}, \tilde{\mathbf{u}}_{0:T-1}, \hat{\theta}, \hat{\omega}))}{\partial \omega} \in \mathbb{R}^{T, m}.$$

Let the residual $\tilde{y}_\ell = \tilde{y} - h(\hat{F}(\hat{\mathbf{x}}_{0:T-1}, \tilde{\mathbf{u}}_{0:T-1}, \hat{\theta}, \hat{\omega})) - \mathscr{J}_\theta \hat{\theta} - \mathscr{J}_\omega \hat{\omega}$, $\Upsilon_\ell = \Upsilon(\mathscr{J}_\theta)$, and assumption 3.3.ii) hold for $\tilde{y}_\ell, \mathscr{J}_\theta, \mathscr{J}_\omega$. If the following conditions hold, i.e.,

$$\|e^\star_{\mathscr{J}_\theta, \mathscr{J}_\omega}(\tilde{y}_\ell, \omega')\|_2 \leq \|e^\star_{\mathscr{J}_\theta, \mathscr{J}_\omega}(\tilde{y}_\ell, \hat{\omega})\|_2, \quad \|\omega'\|_0 \leq M,$$
$$\|e^\star_{\mathscr{J}_\theta, \mathscr{J}_\omega}(\tilde{y}_\ell, \hat{\omega})\|_{(\Upsilon_\ell, 1)} + \|e^\star_{\mathscr{J}_\theta, \mathscr{J}_\omega}(\tilde{y}_\ell, \hat{\omega})\|_{(\Upsilon_\ell, 2M)} < \sigma^2_{2M}(\Upsilon_\ell)\eta(\hat{\omega}),$$

where $\eta(\omega) \doteq \min_{i \in \text{supp}(\omega)} |\omega_i|$, then $\hat{\omega}$ and ω' have the same support $\text{supp}(\hat{\omega}) = \text{supp}(\omega')$ and the same sign $\text{sign}(\hat{\omega}_i) = \text{sign}(\omega'_i)$ for all i.

Proof. The proof follows by linearizing $h(\hat{F}(\cdot))$ around $(\hat{\theta}, \hat{\omega})$ (3.20), i.e.,

$$h(\hat{F}(\cdot)) \simeq h(\hat{F}(\hat{\mathbf{x}}_{0:T-1}, \tilde{\mathbf{u}}_{0:T-1}, \hat{\theta}, \hat{\omega})) + \mathcal{J}_\theta(\theta - \hat{\theta}) + \mathcal{J}_\omega(\omega - \hat{\omega}),$$

applying the proof to Theorem 3.2, thus considering Remark 3.1 and the fact that $\Upsilon_\ell \tilde{y}_\ell = \Upsilon_\ell \mathcal{J}_\omega \hat{\omega} + e^\star_{\mathcal{J}_\theta, \mathcal{J}_\omega}(\tilde{y}_\ell, \hat{\omega})$. □

Next, we define the conditions under which $(\hat{\theta}, \hat{\omega})$ (3.20) is locally maximally sparse within the linearized neighborhood of the solution. Let $\bar{\omega}_\ell$ be the maximally sparse vector in the local approximation around $(\hat{\theta}, \hat{\omega})$, i.e.,

$$\begin{aligned} \bar{\omega}_\ell = \quad & \arg\min_{\theta, \omega} \|\omega\|_0 \\ & \text{s.t. } \|\tilde{y} - \mathcal{J}_\theta \theta - \mathcal{J}_\omega \omega\|_2 \leq \mu. \end{aligned} \tag{3.21}$$

Given $\eta(\omega) \doteq \min_{i \in \text{supp}(\omega)} |\omega_i|$, then we compute ω^v_ℓ as

$$\omega^v_\ell \doteq \arg\min_{\omega \in \mathbb{R}^m} \|\omega\|_1 \tag{3.22a}$$

$$\text{s.t.} \quad \text{sign}(\hat{\omega}_i)\omega_i \geq \eta(\hat{\omega}), \qquad \forall i \in \text{supp}(\hat{\omega}) \tag{3.22b}$$

$$|\omega_i| < \eta(\hat{\omega}), \qquad \forall i \in \overline{\text{supp}}(\hat{\omega}) \tag{3.22c}$$

$$\|\Upsilon_\ell \tilde{y}_\ell - \Upsilon_\ell \mathcal{J}_\omega \hat{\omega}\|_2 \leq \mu. \tag{3.22d}$$

Theorem 3.5 (maximum sparsity local recovery)

Given $\mathcal{D} = \{\tilde{\mathbf{u}}_{0:T}, \tilde{\mathbf{y}}_{0:T}\}$, let Assumptions 3.1, 3.2, 3.3.i) hold. Let $\bar{\omega}_\ell$ and ω^v_ℓ be the solutions of (3.21) and (3.22), respectively. Let $\hat{\omega}$ be the solution of (3.20), with $M \doteq \|\hat{\omega}\|_0$. Define the Jacobian matrices $\mathcal{J}_\theta \in \mathbb{R}^{T, n_\theta}$, $\mathcal{J}_\omega \in \mathbb{R}^{T, m}$, and the residual vector $\tilde{y}_\ell$ as in Theorem 3.4. Let $\Upsilon_\ell = \Upsilon(\mathcal{J}_\theta)$, and let assumption 3.3.ii) hold for $\tilde{y}_\ell, \mathcal{J}_\theta, \mathcal{J}_\omega$. Define $\kappa_e \doteq \|\hat{\omega}\|_0 - \|\bar{\omega}_\ell\|_0$. Consider the set

$$\lambda \doteq \left\{ i \,:\, |\omega^v_{\ell,i}| > \frac{\|e^\star_{\mathcal{J}_\theta, \mathcal{J}_\omega}(\tilde{y}_\ell, \omega^v_\ell)\|_{(\Upsilon_\ell, 1)} + \|e^\star_{\mathcal{J}_\theta, \mathcal{J}_\omega}(\tilde{y}_\ell, \omega^v_\ell)\|_{(\Upsilon_\ell, 2M)}}{\sigma^2_{2M}(\Upsilon_\ell)} \right\}.$$

Assume that the constraint (3.22d) is active, i.e., $\|\Upsilon_\ell \tilde{y} - \Upsilon_\ell \Phi \omega^v\|_2 = \mu$. Then, $\kappa_e \leq \bar{\kappa}_e \doteq \|\hat{\omega}\|_0 - card(\lambda)$. Furthermore, if

$$\frac{\|e^\star_{\mathcal{J}_\theta, \mathcal{J}_\omega}(\tilde{y}_\ell, \omega^v_\ell)\|_{(\Upsilon_\ell, 1)} + \|e^\star_{\mathcal{J}_\theta, \mathcal{J}_\omega}(\tilde{y}_\ell, \omega^v_\ell)\|_{(\Upsilon_\ell, 2M)}}{\sigma^2_{2M}(\Upsilon_\ell)} < \eta(\hat{\omega}),$$

then $\hat{\omega}$ is locally maximally sparse, i.e., $\bar{\kappa}_e = 0$,and $\text{supp}(\hat{\omega}) = \text{supp}(\bar{\omega}_\ell)$.

Proof. The proof follows from Theorem 3.4, following the same procedure adopted in the proof of Theorem 3.3.

Remark 3.3 (Relation with the original problem)**.** *Although the optimization problems in this section are presented in a constrained form, as in* (3.14) *and* (3.20)*, they can always be reformulated in the unconstrained form* (2.6) *within the proposed framework. As discussed in [85], this can be obtained by solving the corresponding Lagrangian problem with appropriate multipliers.*

The conditions established in Theorems 3.3, 3.5 provide a precise characterization of when an estimate attains maximal sparsity, i.e., when no other feasible solution exists with strictly fewer nonzero entries. From a practical viewpoint, these results are valuable because they allow one to certify maximal sparsity without exhaustive search over all possible supports, which would be combinatorially prohibitive, and without requiring prior knowledge of the exact ℓ_0-norm minimization solution of (3.13). The generality highlighted in Remark 3.1 is also significant: the conditions are not tied to a specific optimization program, but rather apply to any vector satisfying the governing equations with a residual term. This means that maximal sparsity can be assessed for estimates obtained through a variety of algorithms, provided the model structure is consistent. Summarizing, this theoretical framework offers both a rigorous and a practical tool for determining whether a given estimate is as sparse as possible under the problem constraints.

These maximal sparsity conditions form a key ingredient for the following analysis, where they are combined with the results of Section 3.1 to prove that the identified physical parameters θ can achieve superior accuracy compared to standard estimates without sparsity guarantees.

3.3 Optimality of the physical parameters

In this section, we show that, under black-box augmentation and maximum sparsity conditions, the identified physical parameters θ are *provably more accurate,* in the worst-case sense, than standard estimates lacking black-box models with sparsity guarantees. Specifically, given a sufficiently rich basis function dictionary φ, the unknown term $\Delta(x_k, u_k)$ in (2.1) with true physical parameters $\bar{\theta}$ can be parameterized as

$$\Delta(x_k, u_k) = \sum_{i=1}^{m} \bar{\omega}_i \varphi_i(x_k, u_k), \tag{3.23}$$

where $\bar{\omega}$ is the maximally sparse coefficient vector solving (3.13). This defines the true system parameters as $(\bar{\theta}, \bar{\omega})$. Thus, we define the parameter sets on θ and ω.

Definition 3.6 (Parameters sets). *Given dataset* $\mathscr{D} = \{\tilde{\mathbf{u}}_{0:T}, \tilde{\mathbf{y}}_{0:T}\}$ *satisfying Assumption 3.2, a state sequence* $\mathbf{x}_{0:T-1}$*, and the noise bound* μ*, recalling* (3.12)*, we have*

$$\text{FPS} \doteq \left\{\theta, \omega \,:\, \|\tilde{y} - \hat{F}(\mathbf{x}_{0:T-1}, \tilde{\mathbf{u}}_{0,T-1}, \theta, \omega)\|_2 \leq \mu\right\}.$$

A subset of FPS, the Supported Feasible Parameter Set (SFPS), is

$$\text{SFPS} \doteq \text{FPS} \cap \left\{\theta, \omega \,:\, \text{supp}(\omega) = \text{supp}(\bar{\omega})\right\},$$

with $\bar{\omega}$ *the maximally sparse coefficients vector.*

The bound μ plays a key role in defining the sets. Thus, we remark that it can be refined through iterative approaches to balance complexity, accuracy, and fit to data. Given a generic estimate $\hat{\theta}$ of the true parameter vector $\bar{\theta}$, we now consider the parametric error defined as $e_\theta(\hat{\theta}) \doteq \|\bar{\theta} - \hat{\theta}\|_p$. Since $\bar{\theta}$ is unknown in practice, the so-called worst-case parametric error, i.e., a tight upper bound on $e_\theta(\hat{\theta})$ over the set of feasible parameters, provides a measure of the maximum possible deviation of $\hat{\theta}$ from $\bar{\theta}$. In a "standard" scenario, i.e., without sparsity guarantees or black-box compensation, the FPS defined in (3.12) is the smallest set that contains $(\bar{\theta}, \bar{\omega})$. Hence, the worst-case parametric error is given by

$$e_\theta^{FPS}(\hat{\theta}) = \sup_{\theta \in \text{FPS}} \|\theta - \hat{\theta}\|_p. \tag{3.24}$$

However, when black-box compensation is applied and $\hat{\omega}$ satisfies the maximum sparsity conditions, SFPS replaces FPS as the minimal guaranteed set, yielding

$$e_\theta^{SFPS}(\hat{\theta}) = \sup_{\theta \in \text{SFPS}} \|\theta - \hat{\theta}\|_p. \tag{3.25}$$

Note that applying (3.25) to an estimate that does not satisfy maximum sparsity conditions is meaningless, as its inclusion in SFPS is not guaranteed. In such cases, FPS remains the smallest valid set. Thus, based on these definitions, the following theorem establishes that finding the "correct sparsity" of the black model leads to a more accurate physical estimate, i.e., an estimate with a lower worst-case parametric error, compared to "standard" estimates without sparsity guarantees or obtained without a black model augmentation.

Theorem 3.6 (maximum sparsity optimality)

Let Assumptions 3.1, 3.2, 3.3.i) hold. Consider Δ as in (3.23), and let $(\hat{\theta}, \hat{\omega})$ be a solution of (3.20) in the region of attraction of $(\bar{\theta}, \bar{\omega})$, for which conditions of Theorem 3.5 hold. Consider any other solution $(\tilde{\theta}, \tilde{\omega})$, for which conditions of Theorem 3.5 do not hold. Then,

$$e_\theta^{\text{SFPS}}(\hat{\theta}) \leq e_\theta^{\text{FPS}}(\tilde{\theta}).$$

Proof. By Theorem 3.5, $\text{supp}(\hat{\omega}) = \text{supp}(\bar{\omega})$, making $\hat{\omega}$ maximally sparse and ensuring $(\hat{\theta}, \hat{\omega}) \in \text{SFPS}$.

On the other hand, being $\widetilde{\omega}$ without sparsity guarantees, it is not ensured that $(\tilde{\theta}, \widetilde{\omega}) \in \text{SFPS}$. Thus, $(\tilde{\theta}, \widetilde{\omega}) \in \text{FPS}$.

From Definition 3.6 we have $\text{SFPS} \subseteq \text{FPS}$. It follows that $e_\theta^{\text{SFPS}}(\hat{\theta}) = \sup_{\theta \in \text{SFPS}} \|\theta - \hat{\theta}\|_p \leq \sup_{\theta \in \text{FPS}} \|\theta - \tilde{\theta}\|_p = e_\theta^{\text{FPS}}(\tilde{\theta})$, showing that "correct sparsity" yields a better worst-case estimate. □

Remark 3.4 (On the benefits of δ)**.** *The result of Theorem 3.6 is closely linked to Theorem 3.1. As evident from* (3.1)*, the better the black-box model compensates for unknown dynamics, the smaller $\widetilde{\Delta}$, thereby tightening the parametric error bound. In particular, note also that, if $\hat{\omega} = \bar{\omega}$, then $\widetilde{\Delta} = 0$.*

3.4 Academic example

Within this section, we validate our theoretical results identifying the nonlinear vehicle lateral dynamics model presented in [87]. The dynamics is defined by

$$\begin{aligned}\tilde{\psi}_{k+1} &= \bar{\theta}^\top [\tilde{\psi}_k,\ \tilde{\psi}_{k-1},\ \tilde{\psi}_k \tilde{p}_k,\ \tilde{s}_{k-1}\tilde{p}_{k-1}]^\top + \Delta_k + d_k,\\ \Delta_k &= -9.625\tilde{\psi}_k\tilde{p}_{k-1} + 10.69\tilde{\psi}_{k-1}\tilde{p}_{k-1} - 11.52\tilde{\psi}_{k-1}\tilde{p}_{k-1}^2,\end{aligned}$$

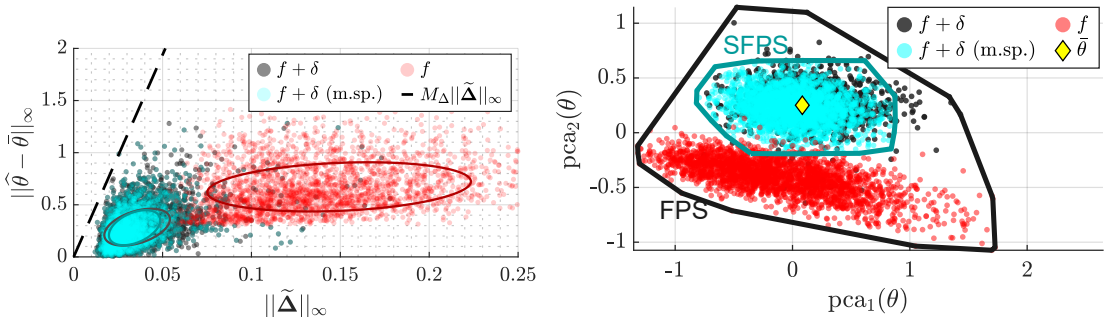
where $\tilde{s}_k$ (steering angle) and $\tilde{p}_k$ (inverse longitudinal velocity) are the measured inputs generated as in [87], and $\tilde{\psi}_k$ is the yaw rate output. The noise term d_k accounts for both process and measurement disturbances. The physical parameters are $\bar{\theta} = [2, -1.087, -1.070, 3.715]^\top$, and Δ_k is the unmodeled dynamic. We recast the known part of the system as a state-space model, i.e.,

$$\begin{aligned}x_{k+1} &= \begin{bmatrix} \theta_1 x_{1,k} + \theta_2 x_{2,k} + \theta_3 x_{1,k} x_{5,k} + \theta_4 x_{4,k} x_{6,k} \\ x_{1,k} \\ u_{1,k} \\ x_{3,k} \\ u_{2,k} \\ x_{5,k} \end{bmatrix} = f(x_k, u_k, \theta),\\ y_k &= x_{1,k} = h(x_k),\end{aligned}$$

with $x_k = [\tilde{\psi}_k, \tilde{\psi}_{k-1}, \tilde{s}_k, \tilde{s}_{k-1}, \tilde{p}_k, \tilde{p}_{k-1}]^\top$, $u_k = [\tilde{s}_k, \tilde{p}_k]$. To capture the unmodeled dynamics, we augment f_1 with $\delta(x_k, u_k, \omega) = \omega^\top \varphi(x_k, u_k)$, where

$$\varphi(x_k, u_k) = [1, x_{3,k}, x_{4,k}, x_{5,k}, x_{6,k}, x_{3,k}^2, x_{4,k}^2, x_{5,k}^2, x_{4,k}x_{6,k}^2, x_{1,k}x_{6,k}, x_{2,k}x_{6,k}, x_{2,k}x_{6,k}^2]^\top.$$

This choice is driven by, e.g., the polynomial composition of the known model f. Thus, the "true" black-box parameter is given by $\bar{\omega} = [0, \ldots, 0, -9.625, 10.69, 11.52]^\top$.



(a) (b)

Figure 3.1: (a) Parametric error with respect to $\|\tilde{\mathbf{\Delta}}\|_\infty$. (b) Approximated FPS and SFPS.

Using $T = 2000$ input-output samples, we identify $\hat{\theta}$ and a sparse $\hat{\omega}$ by minimizing the regularized cost function $\mathscr{C}_T$ (2.5) with $\mathscr{L}_k = \frac{1}{T}\|e_k\|_2^2$ and $\gamma = 0.001$.

First, we performed 7500 simulations – 5000 with the black-box model and 2500 without – under varying noise (with fixed upper bound) and input conditions. Fig. 3.1a shows the parametric errors versus $\|\tilde{\mathbf{\Delta}}\|_\infty$, along with the estimated bound M_Δ and confidence ellipses. The black-box model consistently reduces the error, confirming its effectiveness, while all data points respect the computed bound, validating Theorem 3.1.

Then, Fig. 3.1b shows the approximated FPS and SFPS as convex hulls in the space of the two principal components of θ ($\text{pca}_i(\theta)$). Notably, most parameters identified with δ lie close to the ones in the SFPS, which contains only those for which conditions of Theorem 3.5 hold, yielding $\text{supp}(\hat{\omega}) = \text{supp}(\bar{\omega})$ (see Definition 3.6). This highlights the crucial role of the corrective term δ: by compensating for unmodeled dynamics, it effectively mitigates the bias in the estimation of the physical parameters, resulting in more accurate and consistent estimates. This observation aligns with the results of Theorem 3.1 and Remark 3.4 , which formally establish that the inclusion of δ leads to a systematic reduction of the parametric error bound, thereby enhancing both the accuracy and interpretability of the identified model. In contrast, estimates without δ are clearly biased. A statistic on the worst-case error analysis over all the simulations (mean $\pm$ 1σ) further confirms this. In particular, $e_\theta^{\text{SFPS}}(\hat{\theta})$ is 1.19 ± 0.19 for $\theta \in$ SFPS (3.25), compared to $e_\theta^{\text{FPS}} = 1.79 \pm 0.28$ for $\theta \in$ FPS (3.24), validating Theorem 3.6.

Then, we conducted 5000 simulations for each noise upper bounds $\|\eta^v\|_\infty$, under varying noise and input conditions. Fig. 3.2 shows the parametric error distribution, with box plots illustrating median, interquartile ranges (IQRs), and whiskers extending to 1.5×IQR. The results confirm the linear dependence between error and noise bound, as stated in Theorem 3.1.

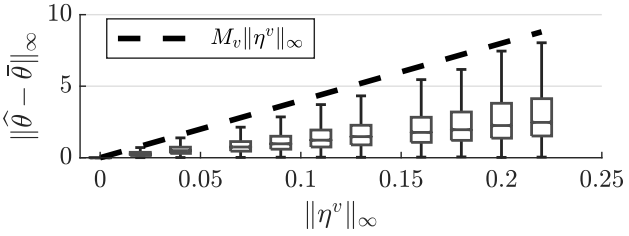


Figure 3.2: Parametric error with respect to $\|\eta^v\|_\infty$.

Discussion and concluding remarks

This chapter has provided the theoretical foundations of the proposed identification framework, establishing formal guarantees on parameter estimation and sparsity recovery. The derived results rigorously quantify the estimation error induced by model uncertainties and approximations, thereby linking identification accuracy to both the data quality and the fidelity of the nominal physical model.

A central message emerging from the presented analysis is the importance of explicitly accounting for modeling uncertainties in system identification. In many practical applications, the available models are inherently simplified representations of the true physical system, often neglecting coupling effects, nonlinearities, or unmodeled disturbances. Neglecting such discrepancies during identification typically forces the estimated parameters to absorb unmodeled effects, resulting in biased and physically inconsistent estimates. Conversely, by formally incorporating unmodeled dynamics into the identification model via δ, the proposed framework provides a principled way to quantify and mitigate these biases, yielding parameter estimates that remain reliable and interpretable even when the underlying physical model is not completely exact.

In this context, it is worth noting that while the proposed regularization scheme promotes sparsity to preserve interpretability, a less stringent regularization of the correction term might be necessary in scenarios where the physical priors are significantly biased or inaccurate. However, as the quality of the prior degrades, the identification of the physical parameters may inevitably be affected. Nonetheless, even in such cases, the correction term remains a valid and effective instrument to mitigate the impact of errors in the priors, preventing them from completely compromising the model's predictive capability. A detailed sensitivity analysis characterizing this trade-off between prior quality, regularization strength, and identification accuracy will be the subject of future works.

From a broader identification perspective, this theoretical treatment reaffirms that the value of an identification method lies not only in its predictive accuracy but also in the credibility and physical soundness of the inferred parameters. This is particularly relevant when simplified models are used for control, fault diagnosis, or digital twin applications, where interpretability and robustness are crucial. The theoretical tools developed in this chapter thus represent a cornerstone of the overall framework,

supporting both its methodological rigor and its practical reliability.

The next chapter builds on these results by extending the proposed identification approach to real-world scenarios characterized by non-uniform observations, demonstrating its robustness and adaptability to incomplete or aggregated data.

Chapter 4

Identification from non-uniform observations

This chapter tackles the issue of identifying system parameters from non-uniform measurements. Numerous practical scenarios require estimation from *irregularly sampled data*, such as when sensors fail, measurements are missing, or only partial runs of the system are available with unknown initial conditions. In these cases, classical identification methods may become unreliable, as they assume regularly spaced data and fully observed trajectories. Here, we extend the framework introduced in Chapter 2 to seamlessly incorporate these non-standard observation settings, while preserving both the interpretability of the physical parameters and the predictive reliability of the identified model.

The chapter is organized around three representative scenarios. Section 4.1 addresses the case of *missing observations*, where measurements are available only at irregular time steps. Section 4.2 considers the situation of *multiple experimental runs*, each with potentially unknown or different initial conditions. Section 4.3 introduces the more challenging case of *aggregated observations*, where measurements are not individual samples but averages collected over a time window, as often occurs in process engineering or biological applications . In each case, we show how the multi-step identification problem defined in Chapter 2 can be reformulated by adapting the cost function or the model, while maintaining the same unifying structure. Section 4.4 presents case studies that illustrate the effectiveness of the proposed extensions.

4.1 Missing observations

We start presenting the scenario of *missing observations*, which arises when only a limited set of output measurements are collected at non-uniform time steps in a given multi-step setting. Practically, this applies to, e.g., sensor failure, data loss, data cleaning, or limited sampling capabilities.

To formally define this setup, as in Chapter 2, we consider a multi-step input/output sequence of length T:

$$\begin{aligned} \tilde{\mathbf{u}}_{0:T-1} &= \{\tilde{u}_0, \dots, \tilde{u}_{T-1}\}, \quad \tilde{u}_k = u_k + v_k, \\ \tilde{\mathbf{y}}_{0:T-1} &= \{\tilde{y}_0, \dots, \tilde{y}_{T-1}\}, \quad \tilde{y}_k = y_k + w_k. \end{aligned} \tag{4.1}$$

Moreover, we define the set of *available time steps* $\boldsymbol{\kappa}_N$, i.e., a set of $N < T$ ordered integers defined as

$$\boldsymbol{\kappa}_N = \{k_1, \dots, k_N\},\ k_j \in [0, T-1],\ \forall j \in [1, N], \tag{4.2}$$

where a time index $k_j \in \boldsymbol{\kappa}_N$ is included if at least one output component is available at time k_j. Accordingly, the sequence of available measurements is defined as follows

$$\tilde{\mathbf{y}}_{[\boldsymbol{\kappa}_N]} = \{\tilde{y}_{k_1}, \dots, \tilde{y}_{k_N}\}. \tag{4.3}$$

In this case, the prediction error term appearing in the cost function $\mathscr{C}_T$ in (2.5), which defines the objective of the multi-step identification problem 2.1, must be evaluated only at those time instants where output measurements are available. Accordingly, for each available measurement index $k_j \in \boldsymbol{\kappa}_N$, the corresponding loss term is defined as

$$\mathscr{L}_{k_j} = \|\tilde{y}_{k_j} - \hat{y}_{k_j}\|_2^2, \tag{4.4}$$

so that missing or unobserved outputs do not contribute to the overall cost. Moreover, we note that the special case of partial measurements, i.e., when only a subset of the output components is available at a given time, can be seamlessly incorporated into the term (4.4). Specifically, this can be done by redefining the loss as $\mathscr{L}_{k_j} = (\tilde{y}_{k_j} - \hat{y}_{k_j})^\top P_{k_j} (\tilde{y}_{k_j} - \hat{y}_{k_j})$, where $P_{k_j} = \mathrm{diag}(p_{11}, \dots, p_{n_y n_y})$ is a diagonal matrix such that, for $i \in [1, n_y]$, $p_{ii} = 1$ if the i-th output component is available at time k_j, and $p_{ii} = 0$ otherwise.

Despite the simplicity of accounting for missing measurements in the optimization problem, it is crucial to assess their effect on the estimation error in the identified parameters, relying on Assumption 3.1.i) on the local identifiability, recalled in the following.

Assumption 4.1 (Local identifiability)**.** *The system is locally identifiable according to, e.g., [82]. In other words, the Hessian of the loss function evaluated in $\theta^\star$ is always positive definite, i.e.,*

$$H \doteq \left.\frac{\partial^2 \mathscr{C}_T(\theta; \cdot)}{\partial^2 \theta}\right|_{\theta=\theta^\star} \succ 0.$$

In the following theorem, we demonstrate that there exists an upper bound on the discrepancy between the parameters identified with N available measurements and those obtained with a complete set of T data. This bound grows proportionally to

the square root of the percentage of missing measurements and inversely proportional to $\sqrt{T}$.

Theorem 4.1 (Error bound with missing measurements)

Let $\theta_T^\star$ represents the vector of identified parameters obtained by solving problem (2.6) using a complete set of T observations. Similarly, let $\theta_N^\star$ denotes the parameters identified using (4.4), i.e., when only $0 < N \leq T$ observations are available, due to missing data. Define $p_{\text{miss}} = \frac{T-N}{T}$ as the percentage of missing observations and let Assumption 4.1 hold. Then, the error between the identified parameters under missing measurements and those from the complete dataset satisfies

$$\|\theta_T^\star - \theta_N^\star\|_2 \leq \sigma_\xi \frac{1}{\sqrt{T}} \sqrt{p_{\text{miss}}}, \tag{4.5}$$

for some constant $\sigma_\xi \in \mathbb{R}$.

Proof. Let us consider a cost function of the form

$$\mathscr{C}(\theta, \gamma) = \gamma^\top \zeta_T, \tag{4.6}$$

where $\gamma \in \mathbb{R}^T$ is a generic vector of real coefficient and $\zeta_T \in \mathbb{R}^T$ is a vector containing the squared ℓ_2-norm of the prediction error for each element , i.e., $\zeta_{T,k} = \|\tilde{y}_k - \hat{y}_k\|_2^2$. Now, since $\mathscr{C}$ depends on γ, we notice that also a minimizer of this cost is a function of γ. Hence, denoting with $\theta^\star = [\theta_1^\star, \dots, \theta_{n_\theta}^\star]$ the solution obtained by minimizing a cost function of the type (4.6), we have also that $\theta^\star \equiv \theta^\star(\gamma)$. Then, given $\gamma_1 \neq \gamma_2$, we define the associated cost functions according to (4.6), i.e.,

$$\mathscr{C}_1(\theta, \gamma) = \gamma_1^\top \zeta_T, \tag{4.7a}$$

$$\mathscr{C}_2(\theta, \gamma) = \gamma_2^\top \zeta_T. \tag{4.7b}$$

Therefore, considering the solutions of (4.7a) and (4.7b), and applying the mean value theorem, for each parameter $\theta_i^\star(\cdot)$ there exists a vector $\breve{\gamma}^{(i)} = (1-a)\gamma_1 + a\gamma_2$, with $a \in [0,1]$, such that

$$\theta_i^\star(\gamma_1) - \theta_i^\star(\gamma_2) = \xi(\breve{\gamma}^{(i)})^\top (\gamma_1 - \gamma_2), \tag{4.8}$$

with $\xi(\gamma) \doteq \frac{\partial \theta_i^\star(\gamma)}{\partial \gamma} \in \mathbb{R}^T$. Then, to compute $\frac{\partial \theta^\star(\cdot)}{\partial \gamma} \in \mathbb{R}^{n_\theta, T}$ we rely on the implicit differentiation technique. First, we consider that the minimizer $\theta^\star$ is a solution of $\frac{\partial \mathscr{C}_T(\theta,\gamma)}{\partial \theta} = 0$. It follows that

$$\frac{\mathrm{d}}{\mathrm{d}\gamma} \frac{\partial \mathscr{C}_T(\theta^\star, \gamma)}{\partial \theta} = 0, \tag{4.9}$$

where $\frac{\mathrm{d}}{\mathrm{d}\gamma}$ denotes the Jacobian with respect to γ, given by

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\gamma}\frac{\partial \mathscr{C}_T(\theta^\star,\gamma)}{\partial\theta} &= \frac{\partial^2 \mathscr{C}_T(\theta^\star,\gamma)}{\partial\gamma\partial\theta} + \frac{\partial^2 \mathscr{C}_T(\theta^\star,\gamma)}{\partial^2\theta}\frac{\partial\theta^\star(\gamma)}{\partial\gamma} \\ &= G(\gamma) + H(\gamma)\frac{\partial\theta^\star(\gamma)}{\partial\gamma},\end{aligned} \tag{4.10}$$

with H the Hessian matrix of the cost function with respect to θ, and G reflecting the influence of missing data on the system's behavior. Therefore, considering (4.9), (4.10), and knowing that the Hessian H is invertible according to Assumption 4.1, we obtain

$$\frac{\partial\theta^\star(\gamma)}{\partial\gamma} = -H^{-1}G(\gamma).$$

This implies that, for the i-th parameter $\theta_i^\star$, the vector $\xi(\check{\gamma}^{(i)})^\top$ in (4.8) corresponds to the i-th row of the matrix $-H^{-1}G(\check{\gamma}^{(i)}) \in \mathbb{R}^{n_\theta,T}$. Thus, according to (4.8), we have

$$\theta^\star(\gamma_1) - \theta^\star(\gamma_2) = \Xi^\top(\gamma_1 - \gamma_2),$$

with $\Xi = \left[\xi(\check{\gamma}^{(1)}), \dots, \xi(\check{\gamma}^{(n_\theta)})\right] \in \mathbb{R}^{T,n_\theta}$, that yields

$$\begin{aligned}\|\theta^\star(\gamma_1) - \theta^\star(\gamma_2)\|_2 &= \|\Xi^\top(\gamma_1 - \gamma_2)\|_2 \\ &\leq \|\Xi^\top\|_2\|(\gamma_1 - \gamma_2)\|_2.\end{aligned} \tag{4.11}$$

Let us now consider the case of missing measurements and a set of available time-steps $\boldsymbol{\kappa}_N$ (4.2). Specifically, let us define the coefficient vectors in (4.7a) and (4.7b) as

$$\gamma_1 = \gamma_T \doteq \left[\frac{1}{T}, \dots, \frac{1}{T}\right]^\top \tag{4.12a}$$

$$\gamma_2 = \gamma_N = \left[\gamma_{N,i}\right],\ \gamma_{N,i} \doteq \begin{cases} 0 & \text{if } i \notin \boldsymbol{\kappa}_N, \\ \frac{1}{T} & \text{otherwise.}\end{cases} \tag{4.12b}$$

Then, for the coefficient vector γ_T, the associated cost function $\mathscr{C}(\theta,\gamma_T)$ of the form (4.6) is equivalent to a cost function of the form (2.5) with no missing observations[a]. Indeed, we have

$$\begin{aligned}\mathscr{C}(\theta,\gamma_T) = \gamma_T^\top\zeta_T &= \left[\frac{1}{T}, \dots, \frac{1}{T}\right]\begin{bmatrix} \|\tilde{y}_0 - \hat{y}_0\|_2^2 \\ \vdots \\ \|\tilde{y}_{T-1} - \hat{y}_{T-1}\|_2^2 \end{bmatrix} \\ &= \sum_{k=0}^{T-1}\frac{1}{T}\|\tilde{y}_k - \hat{y}_k\|_2^2 = \frac{1}{T}\sum_{k=0}^{T-1}\|\tilde{y}_k - \hat{y}_k\|_2^2.\end{aligned} \tag{4.13}$$

[a] extra penalty terms that do not differ when missing measurements are involved are omitted for simplicity.

Analogously, for the coefficient vector ${\gamma_N}^a$ defined in (4.12b), the associated cost function $\mathscr{C}(\theta, \gamma_N)$ of the form (4.6) is equivalent to one given by (4.4), i.e.,

$$\mathscr{C}(\theta, \gamma_N) = \gamma_N^\top \zeta_T = \frac{1}{T} \sum_{j=0}^{N} \|\tilde{y}_{k_j} - \hat{y}_{k_j}\|_2^2, \tag{4.14}$$

where k_j is the j-th element of $\boldsymbol{\kappa}_N$. Hence, according to (4.13) and (4.14), it follows that (4.11) holds for the solutions $\theta_T^\star$ and $\theta_N^\star$ minimizing (2.5) with complete measurements and with the prediction error term defined as in (4.4), respectively. Moreover, defining σ_ξ as the maximum singular value of the matrix $\Xi^\top$, i.e., $\sigma_\xi = \|\Xi^\top\|_2 \doteq \sigma_{max}(\Xi^\top)$, we obtain

$$\|\theta^\star(\gamma_T) - \theta^\star(\gamma_N)\|_2 = \|\theta_T^\star - \theta_N^\star\|_2 \leq \sigma_\xi \|\gamma_T - \gamma_N\|_2. \tag{4.15}$$

Then, observing that

$$\|(\gamma_T - \gamma_N)\|_2 = \sqrt{\frac{T-N}{T^2}} = \frac{1}{\sqrt{T}} \sqrt{p_{\text{miss}}}, \tag{4.16}$$

and combining (4.15) with (4.16), we yield (4.5), concluding the proof. □

[a] Notice that when $N = 0$, i.e., $p_{\text{miss}} = 1$, the system becomes non-identifiable, as $\mathscr{C}_T(\theta, \cdot) = 0$ for all θ, thereby violating Assumption 4.1.

Theorem 4.1 establishes a link between the percentage of missing observations, p_{miss}, the multi-step horizon T, and the error in the identified parameters. This bound indicates that the worst-case parametric error increases with the square root of the missing data fraction, highlighting the sensitivity of parameter estimation to the gaps in the data. On the other hand, the factor $\frac{1}{\sqrt{T}}$ highlights that being the missing data percentage fixed, the datasets collected over shorter horizons (i.e., lower T) are inherently more sensitive to missing data, thus leading to a relatively larger error. At the same time, larger datasets effectively help to mitigate the negative impact of missing entries. The proportionality constant σ_ξ depends on the maximum singular value of a matrix whose columns describe how the optimal parameters vary with respect to data weighting. This suggests that systems with certain structural properties (i.e., small σ_ξ) are more robust to incomplete datasets.

4.2 Multiple runs

We now consider the case of *multiple runs*, where identification relies on data from

different system trajectories. This is the case, for example, of repeated experiments under varying initial conditions, different inputs, environmental disturbances, or sensor placements. Additionally, multiple runs are also exploited to identify a more robust model, capable of capturing a wider range of system behaviors and generalizing well to unseen scenarios. Alternatively, multiple runs can be defined from a single trajectory by dividing it into smaller segments (see, e.g., the multiple shooting method in [68]) to obtain specific optimization properties, such as smoothing the cost function and improving numerical stability. In this setup, we consider M different runs of the system $\mathcal{S}$ (2.1), starting from M different initial conditions. For each i-th experiment, with $i = [1, M]$, the following noise-corrupted sequences of input-output data, each of length[1] T_r, are available, i.e.,

$$\begin{aligned} \tilde{\mathbf{u}}^{(i)}_{[0:T_r-1]} &= \{\tilde{u}^{(i)}_0, \dots, \tilde{u}^{(i)}_{T_r-1}\}, \\ \tilde{\mathbf{y}}^{(i)}_{[0:T_r-1]} &= \{\tilde{y}^{(i)}_0, \dots, \tilde{y}^{(i)}_{T_r-1}\}. \end{aligned} \tag{4.17}$$

Hence, the prediction-error component of the cost function $\mathscr{C}_T$ in (2.5) can be reformulated to explicitly account for the presence of multiple experimental runs. In particular, the cost can be expressed as

$$\mathscr{C}_T = \frac{1}{MT_r} \sum_{i=1}^{M} \sum_{k=0}^{T_r-1} \mathscr{L}^{(i)}_k, \quad \mathscr{L}^{(i)}_k = \|\tilde{y}^{(i)}_k - \hat{y}^{(i)}_k\|_2^2, \tag{4.18}$$

where $\mathscr{L}^{(i)}_k$ represents the squared prediction error at time step k of the i-th run. In the case of multiple runs, the number of decision variables in the optimization problem (2.6) clearly increases. Indeed, since each run starts from a different initial condition $x^{(i)}_0$, $i \in [1, M]$, the optimization problem (2.6) must be minimized with respect to all initial conditions $x^{(1)}_0, \dots, x^{(M)}_0$. Moreover, it is worth noting that missing measurements and multiple runs can occur simultaneously, as remarked in the following.

Remark 4.1 (Multiple runs with missing measurements)**.** *Consider M independent runs, each of nominal length T_r. For the i-th run, let the set of available measurement instants be*

$$\boldsymbol{\kappa}^{(i)}_N = \{k^{(i)}_1, \dots, k^{(i)}_N\},\ k_j \in [0, T_r - 1],\ \forall j \in [1, N],$$

where $N < T_r$ denotes the number of available measurements[2]. The corresponding available output samples are then collected as

$$\tilde{\mathbf{y}}^{(i)}_{[\boldsymbol{k}^{(i)}_N]} = \{\tilde{y}^{(i)}_{k^{(i)}_1}, \dots, \tilde{y}^{(i)}_{k^{(i)}_N}\}.$$

[1]For simplicity, we assume that each trajectory has the same length T_r. However, this non-restrictive assumption can be easily relaxed to accommodate sequences of different lengths.

[2]Without loss of generality, we use the same N and T_r for each run.

Algorithm 1 Identification with multiple runs and missing measurements

Inputs: Non-uniform dataset with missing data (4.3), and multiple runs (4.17). Time indices with available outputs $\boldsymbol{\kappa}_N^{(i)}$ (4.2) for each run. Initial guesses for θ, ω, and $x_0^{(i)}$.
while not converged **do**
    for each run $i = 1$ to M **do**
        Simulate model over T_r using current $(\theta, \omega, x_0^{(i)})$.
        Compute error e_k at times $k \in \boldsymbol{\kappa}_N^{(i)}$.
    end for
    Compute total cost function over all runs (4.18).
    Update θ, ω, $x_0^{(i)}$ using the first-order optimization algorithm in Appendix A.
end while
Output: Estimated parameters $\theta^\star$, $\omega^\star$, and $x_0^{(i)\star}$.

Accordingly, when multiple runs are available but measurements are missing at some time steps, the prediction-error component of the cost function $\mathscr{C}_T$ can be defined by combining (4.4) *and* (4.18)*, yielding*

$$\mathscr{C}_T = \frac{1}{MT_r} \sum_{i=1}^{M} \sum_{j=1}^{N} \mathscr{L}^{(i)}_{k_j^{(i)}}, \quad \mathscr{L}^{(i)}_{k_j^{(i)}} = \| \tilde{y}^{(i)}_{k_j^{(i)}} - \hat{y}^{(i)}_{k_j^{(i)}} \|_2^2.$$

This formulation ensures that only the time instants corresponding to available measurements contribute to the overall loss, while maintaining consistency across multiple experimental runs.

To enhance the clarity of the proposed method, Algorithm 1 summarizes the key steps of the identification procedure in the presence of missing data and multiple runs.

4.3 Aggregated observations

The case of aggregated observations arises when, over a given time window, only collective information from multiple individual measurements is available. This situation typically occurs when the objective is to monitor long-term trends or cumulative variations of certain quantities, rather than capturing high-frequency or instantaneous data, often due to practical or experimental constraints.

In this context, it is useful to distinguish between two forms of aggregation. The first is running averaging, where each observation represents the average of a sequence of preceding measurements, including the current one. The second is periodic averaging, where each aggregated observation corresponds to the mean of a fixed set of measurements collected over a specified time window, typically without overlap between consecutive windows. While the proposed framework can accommodate running averages

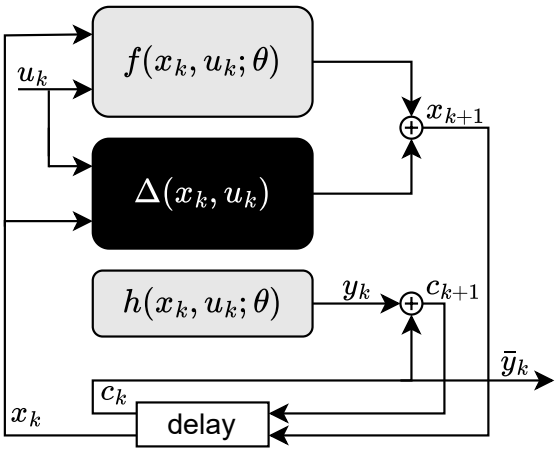


Figure 4.1: Extended system configuration accounting for aggregated observations.

as a particular case, the primary focus in the following will be on periodic averaging.

While missing measurements and multiple runs can be handled with minor adjustments to the cost function (2.5), aggregated measurements require differently refining the initial formulation. Specifically, let us consider T observations and M (possibly consequent) time windows of length $T_r \doteq T/M$ for system (2.1). Then, the sequence of available measurements can be defined as

$$\begin{aligned} \tilde{\mathbf{Y}}_M = \{\tilde{Y}_{T_r}^{(1)}, \dots, \tilde{Y}_{T_r}^{(M)}\}, \quad \tilde{Y}_{T_r}^{(i)} = Y_{T_r}^{(i)} + W_i, \\ Y_{T_r}^{(i)} \doteq \alpha \sum_{k=0}^{T_r-1} y_k^{(i)}, \quad \forall i \in [1, M], \end{aligned} \tag{4.19}$$

with W_i the measurement noise related to the i-th cumulative observation, and the parameter α defined according to the type of data, i.e., $\alpha = 1$ for cumulative measurements and $\alpha = \frac{1}{T_r}$ when considering averaged measurements.

This non-standard representation of the observations compresses multiple individual measurements into a single data point, concealing short-term dynamics and making it difficult to directly apply standard identification techniques. To address this challenge, we propose an extended system model that reinterprets the problem as one involving both missing measurements and multiple runs, as detailed next.

First, we define the following extended system,

$$\bar{\mathcal{S}} : \quad x_{k+1} = f\left(x_k, u_k; \theta\right) + \Delta(x_k, u_k), \tag{4.20a}$$

$$c_{k+1} = c_k + h\left(x_k, u_k; \theta\right), \tag{4.20b}$$

$$\bar{y}_k = \alpha c_k, \tag{4.20c}$$

where $c_k \in \mathbb{R}^{n_z}$ denotes the cumulative state, which integrates or accumulates the output signals $h(x_k, u_k; \theta)$ over time. The corresponding output of the extended system

is denoted by $\bar{y}_k \in \mathbb{R}^{n_z}$. The resulting system configuration is schematically illustrated in Figure 4.1.

The following theorem shows that the case of aggregated measurements can be seamlessly addressed within the same framework developed for missing measurements and multiple runs, by appropriately formulating the identification problem on the extended system $\bar{\mathcal{S}}$ in (4.20).

Theorem 4.2 (Systems equivalence)

Let us consider M aggregated (cumulative or averaged) observations defined by (4.19) for the system (2.1), collected from M (possibly consecutive) time windows of length T_r. Let $x_0^{(i)}$ be the initial condition of the i-th time window. Then, let us consider M multiple runs of length $T_r + 1$ of the system (4.20), having initial conditions $[x_0^{(i)\top}, 0_{n_z}^\top]^\top$. For each run, let us consider missing measurements, as detailed in Section 4.1, with a vector of available time steps $\boldsymbol{\kappa_1} = \{T_r\}$. The resulting sequence of available M observations for the extended system in (4.20), i.e., $\bar{\mathbf{y}}_{[\boldsymbol{\kappa}_1]}^{(i)}$ with $i \in [1, M]$, corresponds to the aggregated observations defined by (4.19) for system (2.1). That is

$$\bar{\mathbf{y}}_{[\boldsymbol{\kappa}_1]}^{(i)} = Y_{T_r}^{(i)}, \quad \forall i \in [1, M]. \tag{4.21}$$

Proof. First, we consider the extended system (4.20) for a generic run[a] i of length $T_r + 1$ with missing measurements defined by $\boldsymbol{\kappa}_1 = T_r$. From (4.3) we have

$$\bar{\mathbf{y}}_{[\boldsymbol{\kappa}_1]} = \{\bar{y}_{k_1}\} = \bar{y}_{T_r}. \tag{4.22}$$

Moreover, from (4.20b)-(4.20c), we obtain

$$\bar{y}_{T_r} = \alpha c_{T_r} = \alpha c_{T_r-1} + \alpha h(x_{T_r-1}, u_{T_r-1}; \theta). \tag{4.23}$$

Iterating (4.20b) backward, we obtain

$$\begin{aligned} c_{T_r-1} &= c_{T_r-2} + h(x_{T_r-2}, u_{T_r-2}; \theta) \\ &= c_{T_r-3} + h(x_{T_r-3}, u_{T_r-3}; \theta) + h(x_{T_r-2}, u_{T_r-2}; \theta) \\ &\vdots \\ &= c_0 + h(x_0, u_0; \theta) + \cdots + h(x_{T_r-2}, u_{T_r-2}; \theta). \end{aligned} \tag{4.24}$$

Now, from (4.23) and (4.24) it follows that

$$\bar{y}_{T_r} = \alpha h(x_0, u_0; \theta) + \cdots + \alpha h(x_{T_r-1}, u_{T_r-1}; \theta),$$

[a] The superscript (i) is omitted for clarity.

having $c_0 = 0_{n_z}$ by definition. Moreover, from (2.1) we have $y_k = h(x_k, u_k; \theta)$, which implies that

$$\bar{y}_{T_r} = \alpha y_0 + \cdots + \alpha y_{T_r-1} = \alpha \sum_{k=0}^{T_r-1} y_k. \tag{4.25}$$

Finally, considering (4.19) and (4.22), we have that (4.25) implies (4.21), which concludes the proof. □

From Theorem 4.2 it follows that cumulative and aggregated measurements can be managed by accounting for both missing measurements and multiple runs for the system in (4.20). Specifically, for each run, only one measurement is available, representing the accumulation of the outputs within the run. Let us define the extended estimation model as

$$\begin{aligned} \bar{\mathscr{M}} : \quad & \hat{x}_{k+1} = f(\hat{x}_k, u_k; \hat{\theta}) + \delta(\hat{x}_k, u_k; \omega), \\ & \hat{c}_{k+1} = \hat{c}_k + h(\hat{x}_k; \hat{\theta}), \\ & \hat{\bar{y}}_k = \alpha c_k. \end{aligned} \tag{4.26}$$

According to Theorem 4.2 and Remark 4.1, and considering M multiple runs and $\boldsymbol{\kappa_1} = \{T_r\}$, the cost function $\mathscr{C}_T$ can thus be redefined as

$$\mathscr{C}_T = \frac{1}{M} \sum_{i=1}^{M} \sum_{j=1}^{1} \mathscr{L}_{k_j}^{(i)} = \frac{1}{M} \sum_{i=1}^{M} \mathscr{L}_{T_r}^{(i)}, \tag{4.27}$$

with $\mathscr{L}_{T_r}^{(i)} = \|\tilde{Y}_{T_r}^{(i)} - \hat{\bar{y}}_{T_r}^{(i)}\|_2^2$ and $\hat{c}_0^{(i)} = 0_{n_z}$ for all $i \in [1, M]$.

Theorem 4.2 suggests a method that streamlines both analysis and implementation by treating aggregated measurements in the same way as missing data and multiple runs, as discussed in Remark 4.1. Indeed, once the data are represented within this framework, previously established algorithms can be applied directly for identification, avoiding the need for non-standard formulations to handle data aggregation in the cost function. A practical implementation of the proposed framework for aggregated observations is provided in Algorithm 2.

Next, we analyze how the use of aggregated observations affects the estimation accuracy of the identified parameters. Similar to the case of missing data, the following theorem establishes the existence of an upper bound on the discrepancy between the parameters identified from aggregated data, collected over windows of length $T_r < T$, and those obtained from complete observations. This bound grows proportionally to $\sqrt{T_r}$, quantifying the degradation in estimation accuracy introduced by temporal aggregation.

Algorithm 2 Identification with aggregated observations

Input: Aggregated outputs $\tilde{Y}_{T_r}^{(i)}$ (4.19) for M time windows of length T_r, input sequences for each window, scaling factor α. Initial guesses for θ, ω, and $x_0^{(i)}$.
while not converged **do**
  for each window $i = 1$ to M **do**
    Simulate extended model (4.26) over horizon T_r.
    Use $\hat{\tilde{y}}_{T_r}$ as predicted cumulative output.
    Compute error $e_{T_r}^{(i)} = \tilde{Y}_{T_r}^{(i)} - \hat{\tilde{y}}_{T_r}^{(i)}$.
  end for
  Compute total cost (4.27).
  Update θ, ω, and $x_0^{(i)}$ using the first-order optimization algorithm in Appendix A.
end while
Output: Estimated parameters $\theta^\star$, $\omega^\star$, and $x_0^{(i)\star}$.

Theorem 4.3 (Error bound with aggregated observations)

Let $\theta_T^\star$ be the vector of identified parameters obtained as the solution to the optimization problem (2.6) when a complete set of T observations is available. Similarly, let $\theta_{T_r}^\star$ represent the vector of identified parameters obtained when the observations are aggregated over a window of length T_r. Let Assumption 4.1 hold. Then, the error between the identified parameters under aggregated observations and those obtained from the complete dataset satisfies

$$\|\theta_T^\star - \theta_{T_r}^\star\|_2 \leq L_\theta \beta_{T_r}, \tag{4.28}$$

for some constant $L_\theta \in \mathbb{R}$, where β_{T_r} is bounded and depends on $\sqrt{T_r}$ as follows

$$\sqrt{T_r} - 1 \leq \beta_{T_r} \leq \sqrt{T_r} + 1. \tag{4.29}$$

Proof. For simplicity and without loss of generality, let us consider the case of $n_z = 1$. The extension to the general case is straightforward. In the following, given a matrix $A \in \mathbb{R}^{m\times n}$, its induced ℓ_p norm is defined as $\|A\|_p = \sup_{x\neq 0} \frac{\|Ax\|_p}{\|x\|_p}$. In particular, for $p = 2$, $\|A\|_2$ corresponds to the spectral norm of A, i.e., the largest singular value of the matrix. Given the prediction error $e_k = \hat{y}_k - \hat{y}_k$, let us consider the vector $\varepsilon = \begin{bmatrix} e_0, \dots, e_{T-1} \end{bmatrix}^\top \in \mathbb{R}^T$. Moreover, consider the following cost function

$$\mathscr{C}(\Gamma, \theta) = p\|\Gamma\varepsilon\|_2^2,$$

with $p \in \mathbb{R}$ a generic constant, and $\Gamma \in \mathbb{R}^{T,T}$ an aggregation matrix of coefficients. Given that $\mathscr{C}$ depends on Γ, any minimizer of the cost function $\mathscr{C}$ will also be a function of Γ. Therefore, $\theta^\star$, i.e., the solution obtained by minimizing $\mathscr{C}$, can be explicitly expressed as

$$\theta^\star \equiv \theta^\star(\Gamma). \tag{4.30}$$

Thus, considering the cost function $\mathscr{C}(\Gamma, \theta)$ when T uniform measurements are available, we have

$$\mathscr{C}_T = \frac{1}{T}\sum_{k=0}^{T-1} \|e_k\|_2^2 = \frac{1}{T}\|\Gamma_T \varepsilon\|_2^2 \doteq \mathscr{C}(\Gamma_T, \theta), \tag{4.31}$$

with $\Gamma_T = \mathbf{I}_T$, the identity matrix of size T. Similarly, considering M aggregated measurements with windows length T_r, as defined in (4.19), and applying (4.25), we can rewrite (4.27) as

$$\begin{aligned}
\mathscr{C}_T &= \frac{1}{M}\sum_{i=1}^{M} \left\| \alpha \sum_{k=0}^{T_r-1} \tilde{y}_k^{(i)} - \alpha \sum_{k=0}^{T_r-1} \hat{y}_k^{(i)} \right\|_2^2 \\
&= \frac{\alpha^2}{M}\sum_{i=1}^{M} \left\| \sum_{k=0}^{T_r-1} (\tilde{y}_k^{(i)} - \hat{y}_k^{(i)}) \right\|_2^2 \\
&= \frac{\alpha^2}{M} \left\| \Gamma_{T_r} \varepsilon \right\|_2^2 \doteq \mathscr{C}(\Gamma_{T_r}, \theta),
\end{aligned} \tag{4.32}$$

with $\tilde{y}_k^{(i)} = y_k^{(i)} + \frac{\eta_i^Z}{T_r}$, $\Gamma_{T_r} = \left[g_1, \dots, g_T\right]^\top$, and

$$g_j = \begin{cases} \left[0_{(j-1)T_r}^\top, 1_{T_r}^\top, 0_{T-jT_r}^\top\right]^\top & \text{if } j \le M, \\ 0_T & \text{otherwise.} \end{cases} \tag{4.33}$$

Thus, being $\mathscr{C}(\Gamma, \theta)$ twice continuously differentiable and the system identifiable according to Assumption 4.1, by applying the implicit function theorem [84] to the gradient function $\nabla_\theta \mathscr{C}(\Gamma, \theta(\Gamma)) = \frac{\partial \mathscr{C}(\Gamma, \theta(\Gamma))}{\partial \theta}$ it follows that $\theta(\Gamma)$ is continuously differentiable, and consequently also Lipschitz continuous, with respect to Γ. Now, let us consider $\theta_T^\star$ and $\theta_{T_r}^\star$, i.e., the minimizers of (2.5) with complete, standard observations and (4.27), respectively. According to (4.30)–(4.32), we have $\theta_T^\star = \theta^\star(\Gamma_T)$ and $\theta_{T_r}^\star = \theta^\star(\Gamma_{T_r})$. Therefore, from Lipschitz continuity it follows that

$$\|\theta_T^\star - \theta_{T_r}^\star\|_2 \le L_\theta \|\Gamma_T - \Gamma_{T_r}\|_2, \tag{4.34}$$

for some Lipschitz constant L_θ. Then, defining β_{T_r} as the maximum singular value of the matrix $\Gamma_T - \Gamma_{T_r}$, we have

$$\|\Gamma_T - \Gamma_{T_r}\|_2 = \sigma_{max}(\Gamma_T - \Gamma_{T_r}) = \beta_{T_r},$$

which, combined with (4.34), yields (4.28). Thus, applying triangle inequality, the following relation holds, i.e.,

$$\begin{aligned}\beta_{T_r} = \|\Gamma_T - \Gamma_{T_r}\|_2 &\leq \|\Gamma_T\|_2 + \| - \Gamma_{T_r}\|_2 \\ &= \|\Gamma_T\|_2 + \|\Gamma_{T_r}\|_2 \\ &= \sigma_{max}(\Gamma_T) + \sigma_{max}(\Gamma_{T_r}).\end{aligned} \tag{4.35}$$

Similarly, applying the reverse triangle inequality

$$\begin{aligned}\beta_{T_r} = \|\Gamma_T - \Gamma_{T_r}\|_2 &\geq |\|\Gamma_T\|_2 - \|\Gamma_{T_r}\|_2| \\ &= |\sigma_{max}(\Gamma_T) - \sigma_{max}(\Gamma_{T_r})|.\end{aligned} \tag{4.36}$$

Here, it is easy to verify that $\sigma_{max}(\Gamma_T) = \sigma_{max}(\mathbf{I}_T) = 1$. Moreover, the following relation holds for all T_r, i.e.,

$$\sigma_{max}(\Gamma_{T_r}) = \sqrt{T_r}. \tag{4.37}$$

In particular, we have that $\sigma_{max}(\Gamma_{T_r}) = \sqrt{\lambda_{max}(\Gamma_{T_r}\Gamma_{T_r}^\top)}$, where

$$\Gamma_{T_r}\Gamma_{T_r}^\top = \begin{bmatrix} g_1^\top \\ \vdots \\ g_T^\top \end{bmatrix} \begin{bmatrix} g_1, \dots, g_T \end{bmatrix} = \begin{bmatrix} g_1^\top g_1 & \cdots & g_1^\top g_T \\ \vdots & \ddots & \vdots \\ g_T^\top g_1 & \cdots & g_T^\top g_T \end{bmatrix}.$$

Here, according to (4.33), it is easy to verify that $g_{i,j} = 0, \forall i > M, \forall j$, and

$$g_i^\top g_j = \begin{cases} 1_{T_r}^\top 1_{T_r} = T_r & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

Defining as $\mathbf{0}_{n,m} \in \mathbb{R}^{n,m}$ the $n \times m$ matrix of all zeros, it follows that

$$\Gamma_{T_r}\Gamma_{T_r}^\top = \begin{bmatrix} T_r\mathbf{I}_M & \mathbf{0}_{M,T-M} \\ \mathbf{0}_{T-M,M} & \mathbf{0}_{T-M,T-M} \end{bmatrix}$$

is a diagonal matrix, where $\lambda_1 = \cdots = \lambda_M = T_r$, $\lambda_{M+1} = \cdots = \lambda_T = 0$, and $\sigma_{max}(\Gamma_{T_r}) = \sqrt{T_r}$, proving the statement (4.37). Thus, from (4.35)–(4.37), we have

$$|1 - \sqrt{T_r}| \leq \beta_{T_r} \leq \sqrt{T_r} + 1,$$

which leads to (4.29) having $T_r \geq 1$, concluding the proof. Notice that $\beta_{T_r} \approx \sqrt{T_r}$ for large T_r. □

Theorem (4.3) establishes an upper bound on the parametric error when using aggregated observations, showing that the error scales with the square root of T_r, i.e., the length of the aggregation window. This implies that larger aggregation windows (or fewer measurements), can lead to greater deviations in the identified parameters. Similar to the case with missing measurements, this behavior emphasizes the effect of non-uniform observation on the parameter estimation accuracy: while aggregating data may be more practical in some scenarios, the resulting effect can mask short-term dynamics, leading to less accurate identification.

4.4 Case studies

In this section, we present case studies to demonstrate and support the efficacy of the proposed framework in handling missing and aggregated observations, respectively. We remark that in all the examples the optimization is carried out using the first-order optimization algorithm described in Appendix A.

4.4.1 Numerical analysis of the upper bound on missing data

To support the result of Theorem 4.1, we demonstrate how missing observations quantitatively affect the parameter identification accuracy by comparing the identification of a second-order linear system with and without missing measurements. The system, presented in [40], is described by the following transfer function

$$g(z) = \frac{(\theta_1 z + \theta_2)}{(z^2 + \theta_3 z + \theta_4)},$$

and in the canonical companion state-space form as

$$\begin{aligned} x_{k+1} &= \begin{bmatrix} -\theta_3 & -\theta_4 \\ 1 & 0 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u, \\ y_k &= \begin{bmatrix} \theta_1 & \theta_2 \end{bmatrix} x_k, \end{aligned}$$

with $\theta_1 = 0.1037$, $\theta_2 = -0.08657$, $\theta_3 = -1.78$, $\theta_4 = 0.9$. First, we excite the system's step response with a perturbation defined by $\mathcal{N}(0,0.01)$, measured over a horizon $T = 104$. Then, we use this signal within the proposed approach to identify the system parameters, accounting for different percentages of missing measurements, i.e., from $p_{\text{miss}} = 0.05$ to $p_{\text{miss}} = 0.95$. To achieve this goal, we minimize a cost function of the form (4.4) using a first-order optimization method. The predicted states and outputs are propagated along the horizon T, while the gradient is computed via automatic differentiation, relying on the approach described in Appendix A.

Figure 4.2 shows the relationship between the percentage of missing observations (p_{miss}) and the ℓ_2 norm of the difference between parameters identified with $p_{\text{miss}}T$

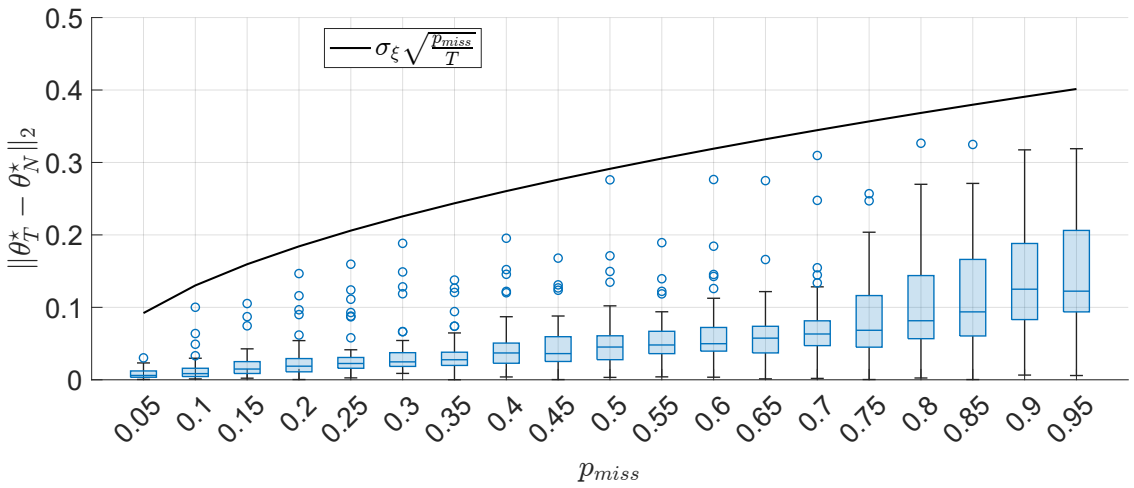


Figure 4.2: Box-plot illustrating the distribution of parameter estimation errors for varying percentages of missing data (p_{miss}).

missing data and in the complete-data case (i.e., $p_{\text{miss}} = 0$), respectively. Each box represents the interquartile range (IQR) of errors, with the median shown as the horizontal line, while the whiskers extend the range of errors to values within $1.5 \times$ IQR ($\approx 3\sigma$) beyond the quartiles. Blue dots represent errors that fall outside this range. The black line represents the upper bound trajectory from Theorem 4.1. Results are collected from 50 simulations for each percentage of missing data, with each simulation featuring different noise values and initial parameter conditions. We can notice that the error bound scales with the proportion of missing data. Moreover, when overlapping the upper bound derived in Theorem 4.1 for $\sigma_\xi = 4.2$ to the numerical data, we can observe that this bound well retrace the data behavior, and it appears more conservative at lower values of p_{miss}. Furthermore, we can observe that the shorter boxes reflect estimation errors tightly clustered around the mean, with few outliers approaching the upper bound. This suggests that, for lower percentages of missing data, the identified parameters are more accurate and stable, exhibiting limited variability even in the presence of some data gaps. Conversely, as p_{miss} increases, the spread of estimation errors around the mean widens, as indicated by the larger boxes. This behavior denotes greater uncertainty and variability in the parameter estimates and it also suggests that a higher amount of missing data makes the identification process less reliable, resulting in a tighter and less conservative upper bound.

To complete the analysis, Figure 4.3 shows the root mean square error (RMSE) between predictions and observations for varying percentages of missing data. In this case, the plot reveals, as expected, an increasing RMSE trend with higher p_{miss} values, thus underscoring the growing discrepancy between predictions and observations as the proportion of missing data rises.

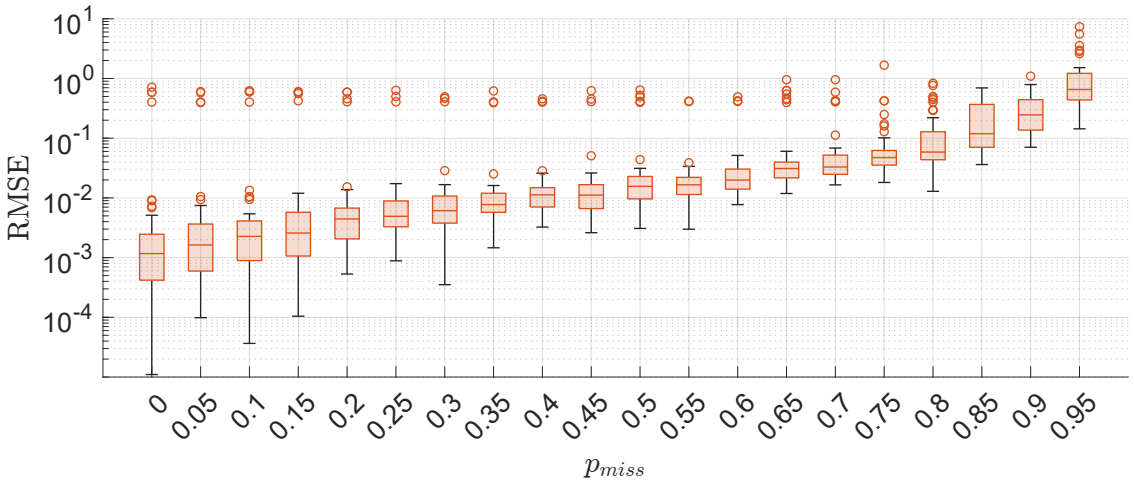


Figure 4.3: Box-plot illustrating the RMSE for varying percentages of missing data (logarithmic scale).

4.4.2 CSTR identification with missing measurements

To explore the practical use of the proposed identification method for handling the case of missing data, we consider the continuous stirred-tank reactor (CSTR) described in [88]. Specifically, we aim to identify the dynamical models for the CSTR relying on a real dataset with different rates of missing observation extracted from the DaISy benchmarks collection [89].

System description

The continuous stirred-tank reactor, depicted in Figure 4.4, is governed by an exothermic process with irreversible reaction, where the product concentration is controlled by regulating the coolant flow. This system has been widely investigated and it is recognized as a highly challenging benchmark for nonlinear process modeling, optimization, and control (see, e.g., [88], [90], [91] and references therein). From a system identification perspective, its inherent nonlinear dynamics and sensitivity to operating conditions make it an ideal test-bench for validating identification strategies. This process has been studied in the literature also in the case of missing data conditions. Some works, such as [52], [92], [93], focus on the same system but rely on different datasets, while others, such as [50], [54], [55], specifically investigate the same dataset used in this paper.

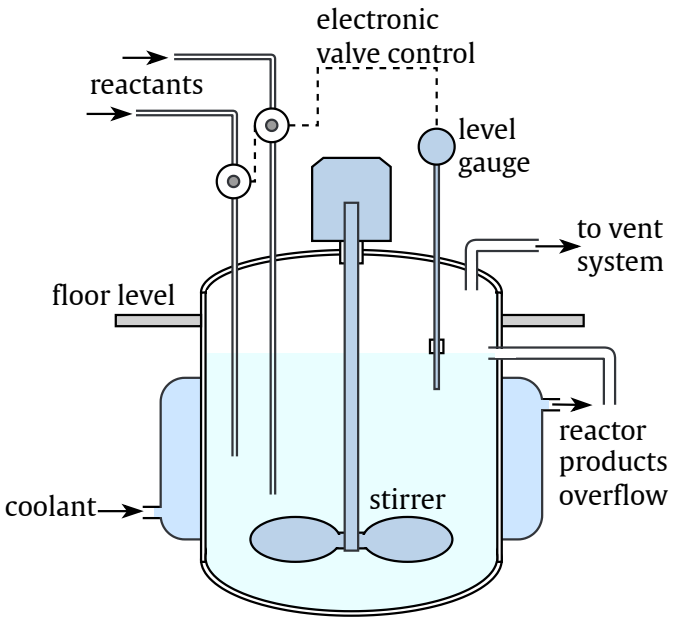


Figure 4.4: Schematic illustration of a CSTR system.

As described in [88], the CSTR system is governed by the following discretized dynamical first-principle equations

$$\begin{aligned} C_{k+1} &= C_k + \Delta t \left[\frac{q}{V}\left(C_0 - C_k\right) - k_0 C_k e^{-\frac{E}{RT_k}} \right], \\ T_{k+1} &= T_k + \Delta t \Big[\frac{q}{V}\left(T_0 - T_k\right) - \frac{(-\Delta H)\, k_0}{\rho C_p} C_k e^{-\frac{E}{RT_k}} \\ &\qquad + \frac{\rho_c C_{pc}}{\rho C_p V} q_{c,k} \left(1 - e^{-\frac{hA}{q_{c,k}\rho C_p}}\right)\left(T_{c0} - T_k\right) \Big], \end{aligned} \tag{4.38}$$

where the product concentration C_k and the reactor temperature T_k are the state variables, whereas the coolant flow rate $q_{c,k}$ is the input. Moreover, in this system, the outputs coincide with the states, i.e., $\tilde{y}_k = \left[C_k, T_k\right]^\top$. The goal is to identify the following vector of parameters $\theta = \left[k_0, \frac{(-\Delta H)k_0}{\rho C_p}, hA\right]^\top$, as in [91] and [52]. The nominal parameter values used in the simulations and their physical description are reported in Table 4.1. The input-output dataset for this process is illustrated in Figure 4.5. It includes 7500 samples, 5000 allocated for the identification task (black line) and the remaining 2500 (red line) reserved for validation.

In the following simulations, we numerically verify Assumption 4.1 by approximating the Hessian using both the Gauss–Newton method (see, e.g., [94]), i.e., $H(\theta) \approx J^\top J$

Table 4.1: Nominal CSTR parameter values.

Name	Description	Value	Unit
C_A	product concentration	x_1	[mol/l]
T	reactor temperature	x_2	[K]
q_c	coolant flow rate	u	[l/min]
q	process flow rate	100	[l/min]
C_0	feed concentration	1	[mol/l]
T_0	feed temperature	350	[K]
T_{c0}	inlet coolant temp	350	[K]
V	CSTR volume	100	[l]
hA	heat transfer term	$7 \cdot 10^5$	[cal/min/K]
k_0	reaction rate constant	$7.2 \cdot 10^{10}$	[min^{-1}]
$\frac{E}{R}$	activation energy term	$1 \cdot 10^4$	[K]
ΔH	heat of reaction	$-2 \cdot 10^5$	[cal/mol]
ρ, ρ_c	liquid densities	$1 \cdot 10^3$	g/l
C_p, C_{pc}	specific heats	1	[cal/g/K]
Δt	sampling time	0.1	[min]

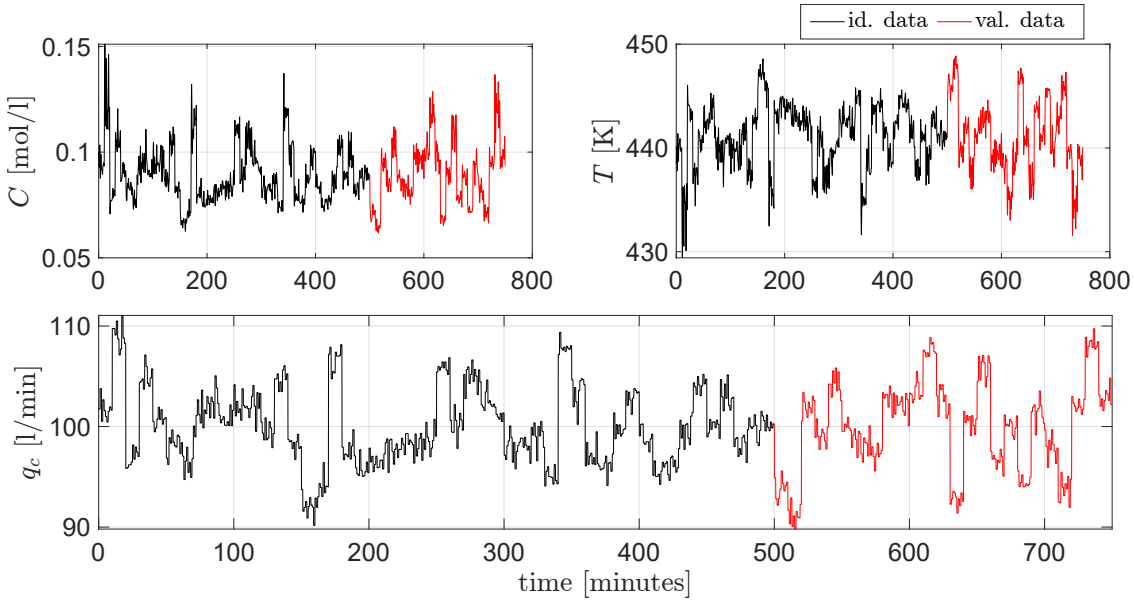


Figure 4.5: Identification (black) and validation (red) input–output measurements related to the CSTR system.

where J is the Jacobian of the residual vector with respect to the parameters and finite-difference perturbations of the gradient. In both cases, the approximate Hessians are positive definite. The Hessian exhibits a condition number $\approx 10^8$, primarily due to

the large variation in the scale of the estimated parameters, which ranged from 10^5 to over 10^{13}. While such disparities in parameter magnitudes can lead to ill-conditioned Hessians, the optimization process is stable and convergent over al the simulations, confirming the practical validity of Assumption 4.1.

Identification results

Identification is achieved by minimizing a cost function of the form (4.4) over the identification data, generating different versions of the original dataset with missing measurement rates ranging from $p_{\text{miss}} = 0\%$ to $p_{\text{miss}} = 75\%$. For each rate of missing data, the results are collected over 200 simulations, each one employing different initial values for the parameters to be identified. Specifically, the estimated parameters are initialized randomly, with each initial value $\hat{\theta}_{i,0}$ selected within a ball around the nominal parameter value θ_i and a radius of 30% of θ_i. That is, $\hat{\theta}_{i,0}$ is chosen such that $\hat{\theta}_{i,0} \in [\theta_i - 0.3\theta_i, \theta_i + 0.3\theta_i]$. The states initial conditions to be estimated are initialized at $\hat{x}_{0,0} = \tilde{z}_0$.

The black-box compensation term $\delta(\cdot)$ is introduced into the dynamical model to handle the process nonlinearities and to guarantee adaptation to unmodeled variations in system parameters. Indeed, the CSTR system is typically subject to changes in the environmental and operational conditions, and it may experience fluctuations that the basic physics-based model (4.38) alone cannot capture accurately [88], [92]. In this example, δ is defined as a linear combination of sigmoid, softplus, hyperbolic tangent, trigonometric, and polynomial functions. Then, a regularization term is introduced in the cost function (4.4) to promote a sparse black-box component. This is done by minimizing an approximation of the ℓ_1-norm of the black-box weights ω (see Chapter 2 for further details). The results presented next are computed on the validation dataset.

Figure 4.6 depicts the effect of missing observations on the model fitness scores, which are computed for the i-th output as

$$\text{fit}_{\%}^{(i)} = 100\left(1 - \frac{\sum_{k=0}^{T-1}(\hat{z}_k^{(i)} - \tilde{z}_k^{(i)})^2}{\sum_{k=0}^{T-1}(\hat{z}_k^{(i)} - \frac{1}{T}\sum_{k=0}^{T-1}\tilde{z}_k^{(i)})^2}\right).$$

The box plot highlights the distribution of the model fitness across 200 simulations for each output and for each level of data loss. Each box shows the IQR, with the median as a horizontal line and whiskers extending to 1.5×IQR. Asterisks represent fitness scores that fall outside this range. The global fitness (yellow boxes) represents the average between the two outputs' fitness, i.e., $\text{fit}_{\%}^{(1)}$ (green boxes) for the first output and $\text{fit}_{\%}^{(2)}$ (red boxes) for the second one. Hence, we have $\text{fit}_{\%} = (\text{fit}_{\%}^{(1)} + \text{fit}_{\%}^{(2)})/2$.

These results demonstrate the robustness of the proposed approach in the case of missing data. Indeed, as the percentage of missing data increases, the average fitness scores remain consistently high, with a gradual decrease only at higher levels of data

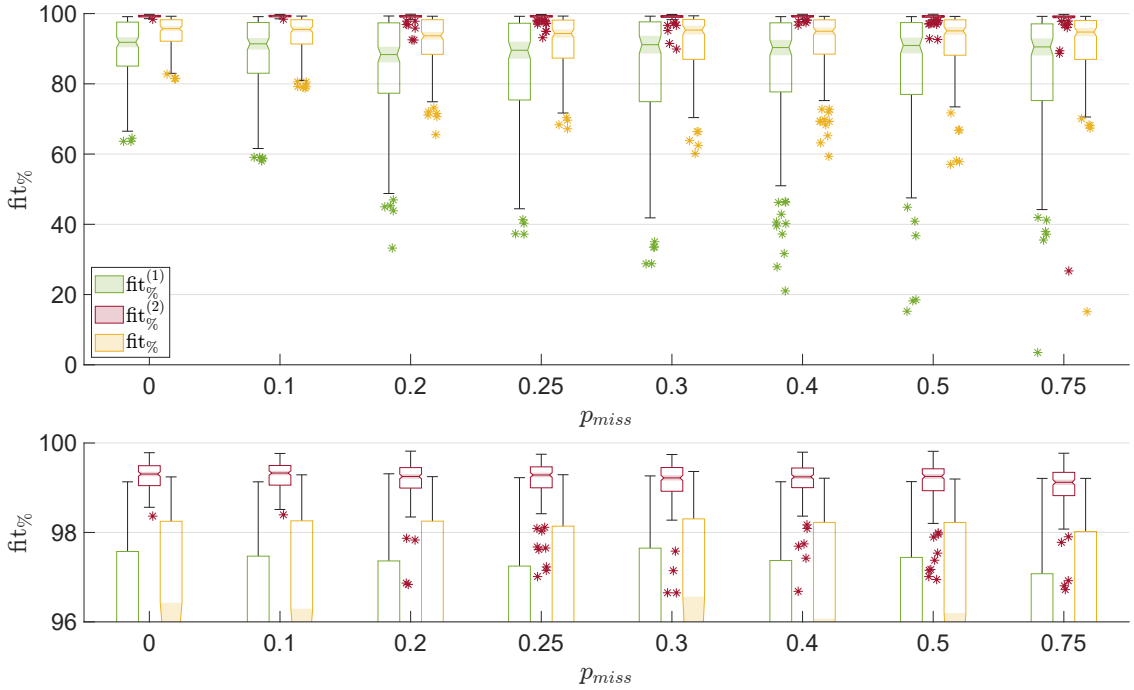


Figure 4.6: Box-plot illustrating the distribution of the fitness scores for varying percentages of missing data. The bottom plot presents a zoom on the fitness scores for the second output specifically.

loss. Moreover, the narrow IQR across simulations for lower values of p_{miss} indicates low variability in the identified models. As p_{miss} increases, the IQR becomes wider, reflecting the expected growth in variability due to reduced information availability. Within this context, we remark that robustness does not imply invariance to missing data but rather refers to the ability of the framework to maintain high average performance and reconstruct meaningful system behavior despite increasing p_{miss}. This behavior is indeed consistent with the theoretical error bounds, which grow with $\sqrt{p_{\text{miss}}}$: as the number of available measurements decreases, errors in parameter estimation naturally increase (see Theorem 4.1), leading to greater variability in prediction quality. Notably, the second output retains better fitness overall, as highlighted in the zoomed section, suggesting a higher resilience of this output to missing observations.

These outcomes are also reflected in Table 4.2, which presents the global fitness scores, compared to the results obtained in [50] (Nuc-SId) and [52] (PF-NSId) for the same amount of missing data. In this case, the results confirm the ability of the approach in maintaining high global fitness scores across varying levels of missing data, outperforming benchmark results obtained with linear (Nuc-SId) and black-box based (PF-SId) identification methods.

Then, the true and predicted trajectories of the CSTR system under four different percentages of missing observations p_{miss} are compared in Figure 4.7, where we have

Table 4.2: Global fitness scores.

p_{miss}	$\text{fit}_{\%}$ (mean $\pm 1\sigma$)	Nuc-SId	PF-NSId
0	94.3 ± 4.84	84.7	89.0
10	93.7 ± 5.47	86.2	88.0
20	91.9 ± 7.49	85.3	87.0
25	91.8 ± 7.61	/	/
30	91.4 ± 8.62	85.4	/
40	91.8 ± 8.40	85.2	/
50	92.1 ± 8.28	83.7	/
75	91.2 ± 9.70	/	/

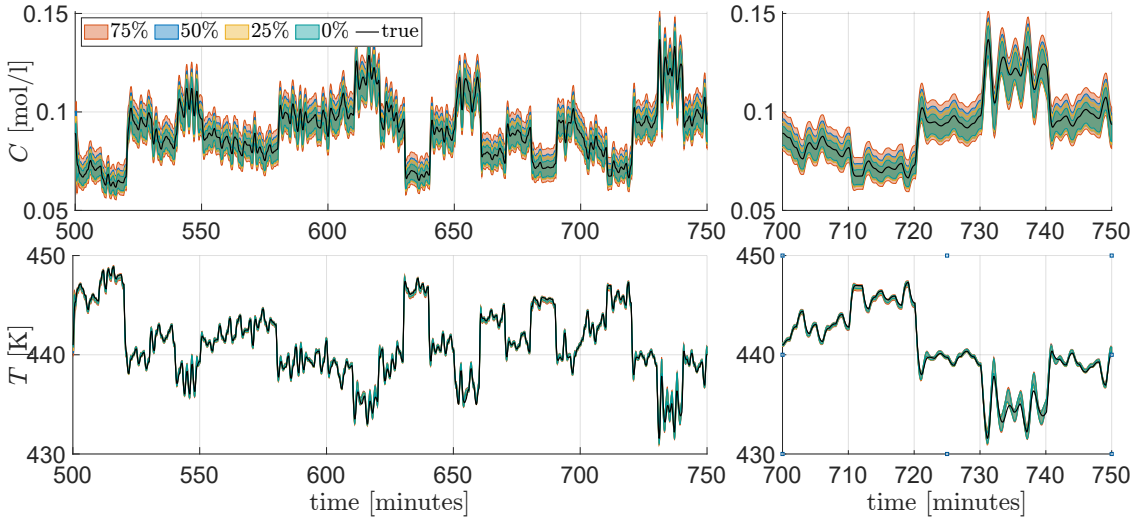


Figure 4.7: Comparison of true and predicted trajectories for the CSTR system under varying percentages of missing data (indicated in the legend), represented with ± 1 standard deviation bands around the mean trajectories. The plots on the right provide a zoomed-in view of the interval [700, 750] minutes to highlight detailed behavior.

also reported the ± 1 standard deviation bands around the mean trajectories. The results highlight the ability of the proposed method to approximate the system's dynamics and confirm its inherent robustness to substantial missing data.

Table 4.3 collects the RMSE scores for the two outputs. The reported values demonstrate the ability of the proposed framework to maintain low the prediction errors across varying levels of missing data for both the outputs, while the relatively small standard deviations indicate stable performance, even under significant data loss. Then, Table 4.4 shows a comparison with [55] (ODE-RSSM), and [54] (NSM-SId) in terms of the relative

Table 4.3: RMSE scores (mean $\pm 1\sigma$).

p_{miss}	$\text{RMSE}_C \times 10^3$	RMSE_T
0	4.54 ± 2.54	0.29 ± 0.06
10	4.88 ± 2.85	0.29 ± 0.06
20	5.61 ± 3.81	0.31 ± 0.09
25	5.80 ± 3.97	0.31 ± 0.10
30	6.09 ± 4.76	0.32 ± 0.10
40	5.94 ± 4.96	0.31 ± 0.07
50	5.76 ± 4.79	0.31 ± 0.09
75	6.39 ± 7.60	0.34 ± 0.16

Table 4.4: RRSE scores.

p_{miss}	RRSE (mean $\pm 1\sigma$)	ODE-RSSM	NSM-SId (mean $\pm 1\sigma$)
0	0.0787 ± 0.0161	0.0659	0.0220 ± 0.005
10	0.0775 ± 0.0152	/	/
20	0.0804 ± 0.0173	/	/
25	0.0785 ± 0.0161	/	/
30	0.0839 ± 0.0192	/	/
40	0.0825 ± 0.0166	/	/
50	0.0834 ± 0.0187	0.1336	0.0920 ± 0.0140
75	0.0864 ± 0.0249	0.2595	/

root squared error (RRSE), calculated as in [54],

$$\text{RRSE} = \frac{1}{n_z} \sum_{i=1}^{n_z} \sqrt{\left(\frac{\sum_{k=0}^{T-1} (\hat{z}_k^{(i)} - \tilde{z}_k^{(i)})^2}{\sum_{k=0}^{T-1} (\hat{z}_k^{(i)} - \frac{1}{T} \sum_{k=0}^{T-1} \tilde{z}_k^{(i)})^2} \right)}.$$

Here, an RRSE of 1 indicates performance equivalent to predicting the mean. For consistency with [55], and [54], the outputs and predictions were normalized in this phase by subtracting their mean and dividing by their standard deviation before calculation. In this case, the results reported in Table 4.4 highlight how the proposed approach is able to provide RRSE values comparable with the ones obtained with the black-box-based benchmark. In particular, better performance are obtained under moderate levels of missing data. Moreover, compared to black-box methods, the proposed approach offers improved interpretability of the identified parameters.

Last, the parameters identified for varying percentages of missing data are reported in Table 4.5. Here, the results reflect the level of accuracy and consistency of the parameter estimation achievable under different levels of data loss, demonstrating the ability of the framework to accurately recover interpretable system parameters, maintaining

Table 4.5: Identified parameters (mean $\pm 1\sigma$).

p_{miss}	$\theta_1 \times 10^{-10}$	$\theta_2 \times 10^{-13}$	$\theta_3 \times 10^{-5}$
0	7.64 ± 0.40	-1.44 ± 0.13	7.00 ± 0.57
10	7.64 ± 0.37	-1.45 ± 0.12	6.93 ± 0.50
20	7.64 ± 0.40	-1.44 ± 0.13	6.97 ± 0.55
25	7.64 ± 0.43	-1.44 ± 0.12	6.88 ± 0.55
30	7.60 ± 0.43	-1.42 ± 0.14	6.99 ± 0.51
40	7.58 ± 0.40	-1.43 ± 0.13	7.01 ± 0.52
50	7.59 ± 0.46	-1.42 ± 0.13	6.91 ± 0.55
75	7.54 ± 0.42	-1.41 ± 0.14	6.92 ± 0.49
Nominal	7.20	-1.44	7.00

relative reliability across different data loss scenarios. However, it is also important to highlight how the identified parameter may differ from the nominal one, as in the case of θ_1. In the considered case study, this discrepancy may be caused by the variations due to environmental and operational conditions, as it commonly happens in real systems.

Summarizing, the application of the proposed approach to the CSTR system in the presence of missing measurements demonstrates its robustness in handling real-world scenarios with substantial missing data. Despite the inherent challenges, the combination of the physics-based model and the black-box component effectively compensates for missing data, accurately identifying the system parameters. Moreover, the obtained results align with those reported in the literature, and a comparison with methods applied to the same benchmark showcases competitive performance particularly under moderate data loss, highlighting the framework's reliability in practical process modeling and its adaptability to real-world conditions.

4.4.3 Identification with averaged observations

In this section, we focus on validating the efficacy of the proposed framework in the case of aggregated observations. In particular, we aim to identify a generic Lotka-Volterra model, which has been largely used to describe the dynamics of a variety of real-world systems.

System description and motivations

The Lotka-Volterra model consists of a set of nonlinear equations commonly used to describe the dynamics of systems involving different interacting species. Specifically, this model describes how the population of the different species varies over time. This model is well-known for describing the dynamics of biological systems, as the interaction between predator and prey populations [44]. However, its application extends also beyond the ecological domain, as for instance in the economic context, where it

is used to represent the wealth of individual investors or the market capitalization of companies [95].

In both ecological and economical framework, the use of averaged measurements is a common practice. For example, significant biological species fluctuations may occur over the year, and sampling on a specific date or during a short period might yield a distorted view of the population's typical behavior [44]. On the other hand, high-frequency data might be unavailable in the economic context, and only averaged values over extended periods can be observed [46]. These averages reflect general trends while concealing short-term variations. Consequently, using aggregated or averaged data within the proposed framework enables the identification of models that fit the available data well, while still capturing detailed dynamics and adapting to the limited resolution of the observations.

Dynamical model

The discretized Lotka-Volterra model with $n_x = 2$ states $x_k = [x_{1,k}, x_{2,k}]^\top$ and parameters $\theta = [\theta_1, \theta_2, \theta_3, \theta_4]^\top$ is given by

$$\begin{aligned} x_{1,k+1} &= x_{1,k} + \theta_1 x_{1,k} - \theta_2 x_{1,k} x_{2,k} + \Delta_1(x_k), \\ x_{2,k+1} &= x_{2,k} - \theta_3 x_{2,k} + \theta_4 x_{1,k} x_{2,k} + \Delta_2(x_k), \\ y_k &= x_k, \end{aligned} \tag{4.39}$$

where x_1 is the population density of prey, x_2 is the population density of the predator, θ_1 and θ_2 are the prey's parameters, describing the maximum per capita growth rate and the effect of predators on the prey growth rate, respectively, θ_3 and θ_4 are the predator's parameters, describing the per capita death rate and the effect of prey on the predator's growth rate, respectively. All parameters are positive and real. On the other hand, Δ_1 and Δ_2 represent unmodeled dynamics that capture external factors affecting the populations beyond the basic predator-prey interaction. In the considered case study we have $\theta = [0.13, 0.02, 0.12, 0.02]$, while Δ_1 and Δ_2 are quadratic terms that may represent, e.g., the intraspecific competition within each population, implying that the growth of each population is influenced not only by the interaction between predator and prey but also by the density-dependent effects within each population. In particular, we considered

$$\Delta_1(x_k) = \mu 10^{-4} x_{1,k}^2, \quad \Delta_2(x_k) = -\mu 5 \cdot 10^{-4} x_{2,k}^2,$$

where $\mu > 0$ is a tuning parameter to control the size of the unmodeled terms. In this case study, $\mu = 10$ is selected to introduce a meaningful but interpretable model mismatch, preserving the structure of the physical model while sufficiently challenging

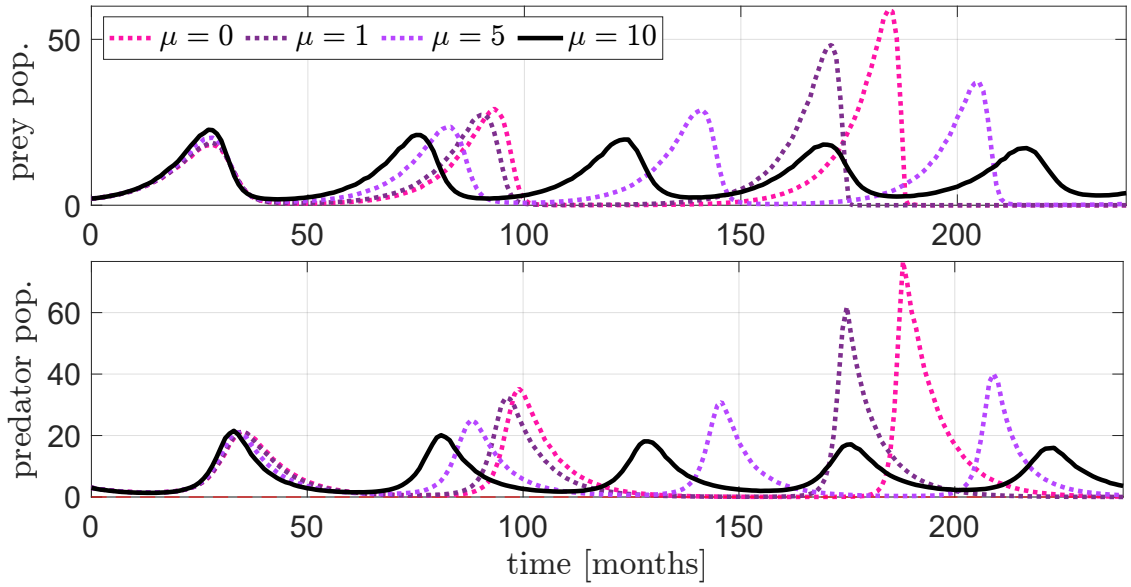


Figure 4.8: Populations evolution for different values of μ.

the black-box compensator[3]. Clearly, the unmodeled dynamics, being unknown, cannot be incorporated into the physical model. Instead, it must be compensated by the black-box approximator δ. Notice that, although the unmodeled dynamics seem relatively small, the impact on the population dynamics is relevant. This is highlighted in Figure 4.8, where the evolution of predator and prey populations is represented over a period of 20 years for different values of μ. In this case, it is evident the importance of efficiently compensating for unmodeled dynamics in order to accurately capture the system's behavior.

As in the previous case study, we conduct a numerical verification of Assumption 4.1 using the Gauss–Newton approximation and the finite-difference methods. The resulting Hessians are positive definite, with condition numbers around 10^3, indicating good numerical conditioning.

Identification results

In the proposed example, we simulate the evolution of the predator and prey populations based on the Lotka-Volterra model (4.39) over 75 years (900 months) with initial condition $x_0 = [2,3]^\top$. The data are generated for both populations at *monthly intervals*, capturing the interaction dynamics described by the model. The first 600 months are used as identification data, while the subsequent 300 months are used for validation. In the considered scenario, the measures averaged over time windows of $T_r = 12$ months

[3]Larger values of μ were found to excessively distort the system dynamics, compromising interpretability and realism at the base of the proposed case study.

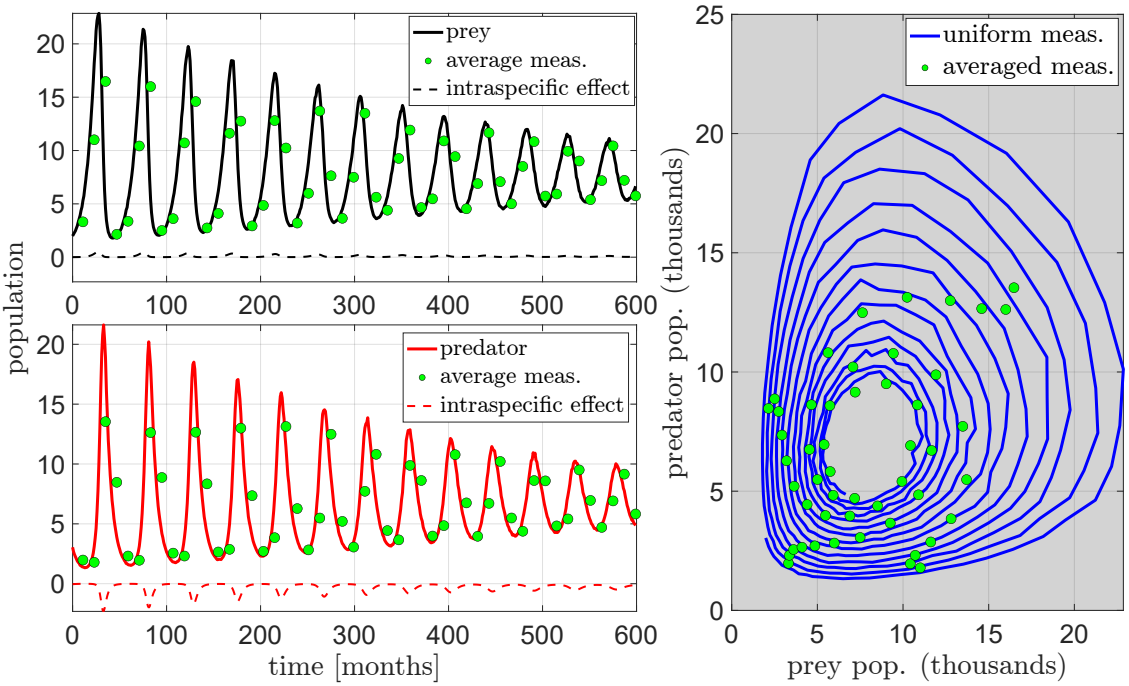


Figure 4.9: Monthly prey (black lines) and predator (red lines) populations evolutions with yearly average measurements (green circles). Dashed lines represent the unmodeled dynamics. On the right, the phase plot of the system with uniform measurements (blue lines) and average measurements is shown.

are exploited, leading to $M = 50$ identification measurements for each population. Figure 4.9 illustrates the monthly population evolution and the yearly average evolution of the prey and predator populations over the entire 50-year identification period. This figure highlights how data averaging captures the overall trend while masking finer details and short-term interactions, which can pose challenges for accurately identifying the system's underlying dynamics.

First, we exploit the results of Theorem 4.2 to identify the system by employing an extended model of the form

$$\begin{aligned}
\hat{x}_{1,k+1} &= \hat{x}_{1,k} + \theta_1 \hat{x}_{1,k} - \theta_2 \hat{x}_{1,k} x_{2,k} + \delta_1(\hat{x}_k), \\
\hat{x}_{2,k+1} &= \hat{x}_{2,k} - \theta_3 \hat{x}_{2,k} + \theta_4 \hat{x}_{1,k} \hat{x}_{2,k} + \delta_2(\hat{x}_k), \\
\hat{c}_{k+1} &= \hat{c}_k + \hat{x}_k, \\
\hat{\bar{y}}_k &= \frac{1}{T_r} \hat{c}_k,
\end{aligned}$$

to estimate the underlying parameters of the predator-prey system from averaged observations. Then, the identification task is performed by minimizing a cost function of the form (4.27) considering M multiple runs and $\boldsymbol{\kappa}_1 = \{T_r\}$.

Table 4.6: Effect of the averaging window size T_r on identification performance.

T_r	$M_{\text{tr}}/M_{\text{val}}$	RMSE_{tr}	RMSE_{val}	$\|\theta - \hat{\theta}\|_2$	$\|x_0 - \hat{x}_0\|_2$
12	50/25	0.2737	0.3436	0.0023	0.2721
15	40/20	0.2183	0.2846	0.0029	0.2047
20	30/15	0.4742	0.5486	0.0123	0.5718
24	25/13	0.8443	0.8502	0.0217	0.9885
40	15/8	1.4328	1.4242	0.0442	1.1534
50	12/6	1.3346	1.3539	0.0589	0.9157

The estimated parameters are initialized randomly as $\hat{\theta}_0 = \theta + \mathcal{N}(0.02, \sigma_\theta)$, with $\sigma_\theta = 0.05$. Analogously, the initial conditions of the states are initialized at $\hat{x}_{0,0} = \tilde{y}_0$. Also in this case, the black-box term δ is defined as a linear combination of selected basis functions, i.e., sigmoid, softplus, hyperbolic tangent, and trigonometric functions. Additionally, the cost function $\mathscr{C}_T$ incorporates physical penalties and regularization terms to enforce specific properties. Specifically, the positivity of $\hat{\theta}$ is ensured using an exponential barrier function, while the sparsity of the black-box component δ is promoted through an ℓ_1-norm approximation applied to the black-box weights ω [69].

Figure 4.10 and Figure 4.11 showcase the predictions from the identified model compared with the averaged observations and the population behavior for identification and validation data, respectively. The results highlight how the proposed approach is able to successfully reconstructs the predator and prey dynamics based on the available averaged data and demonstrate the accuracy of the identified model even when facing aggregated measurements.

The accuracy of the method is also reflected in to adherence of the identified parameters and the state initial condition, i.e., $\hat{\theta} = [0.1318,\ 0.0204,\ 0.1214,\ 0.0198]^\top$ and $\hat{x}_0 = [2.0842, 2.7412]^\top$, with the ground truth values used in the simulation, i.e., $\theta = [0.13, 0.02, 0.12, 0.02]$ and $x_0 = [2,\ 3]^\top$.

Next, we extend the preliminary analysis by considering different sizes of averaging windows. Hence, in addition to the $T_r = 12$-month window (yielding $M = 50$ averaged data points), we fixed the total amount of data ($T = 900$) and the starting value of the estimated parameters and initial condition. Then, we analyze the identification outcomes using larger windows T_r, which imply a lower number of available average measurements. Table 4.6 summarizes the results for each simulated case, including the prediction accuracy measured in terms of the root mean square error[4], and parametric error for both θ, for each window size.

From the reported results, we can observe that, being $T = MT_r$ fixed, an increase

[4]The RMSE has been computed considering the entire set of observations and predictions over the horizon.

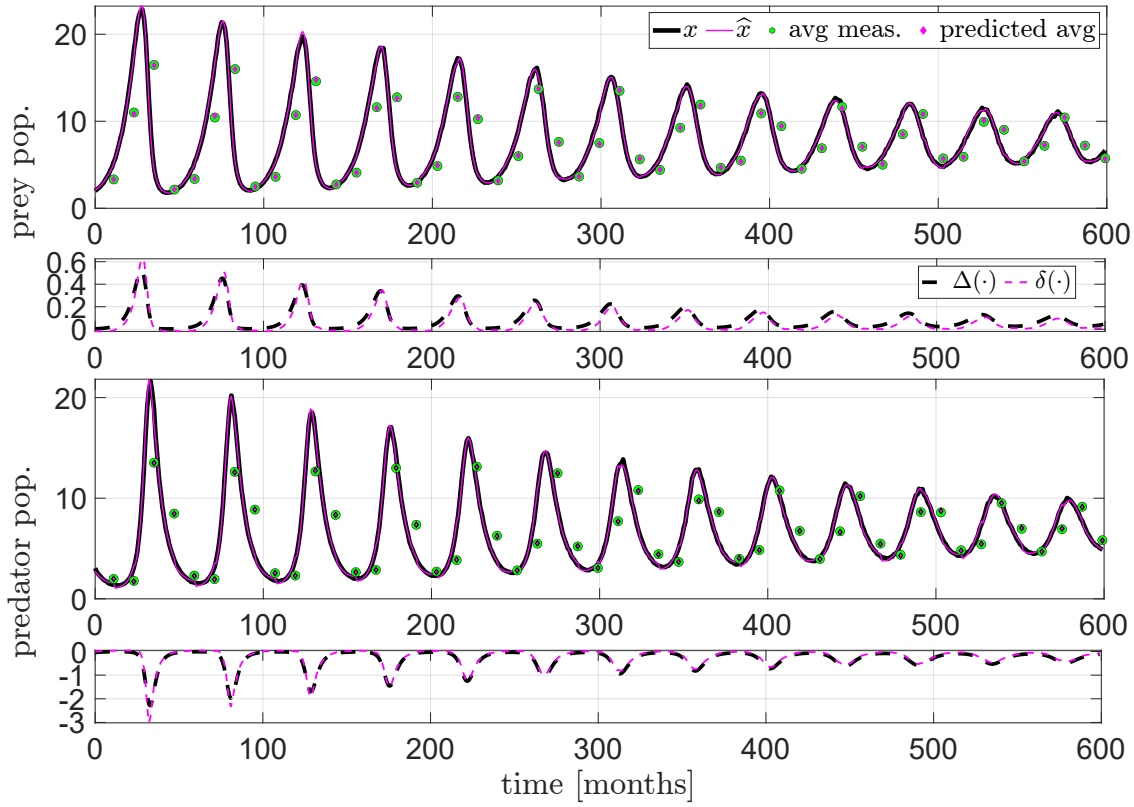


Figure 4.10: Comparison between the true evolution of the populations (black lines) and the one predicted (purple lines) using the model identified from averaged measurements on the identification data. Green markers represent the averaged measurements, while purple markers represent the reconstructed averages. Purple dashed lines indicate the unmodeled dynamics predicted by the black-box term $\delta(\cdot)$, compared with the true one, $\Delta(\cdot)$ (black dashed line).

in T_r results in fewer available measurements (M), which generally leads to less accurate estimates of the parameters and initial conditions, as indicated by increasing error values. Furthermore, a larger T_r not only decreases the data available for identification but also enhances the smoothing effect on short-term dynamics, due to averaging over more values. Consequently, this implies a reduced accuracy, as reflected in the observed trends. Last, it is worth noting that the similar error values observed for $T_r = 12$ and $T_r = 15$ suggest a range where the performance remains relatively stable, indicating the presence of an "optimal" averaging window size, beyond which performance begins to degrade.

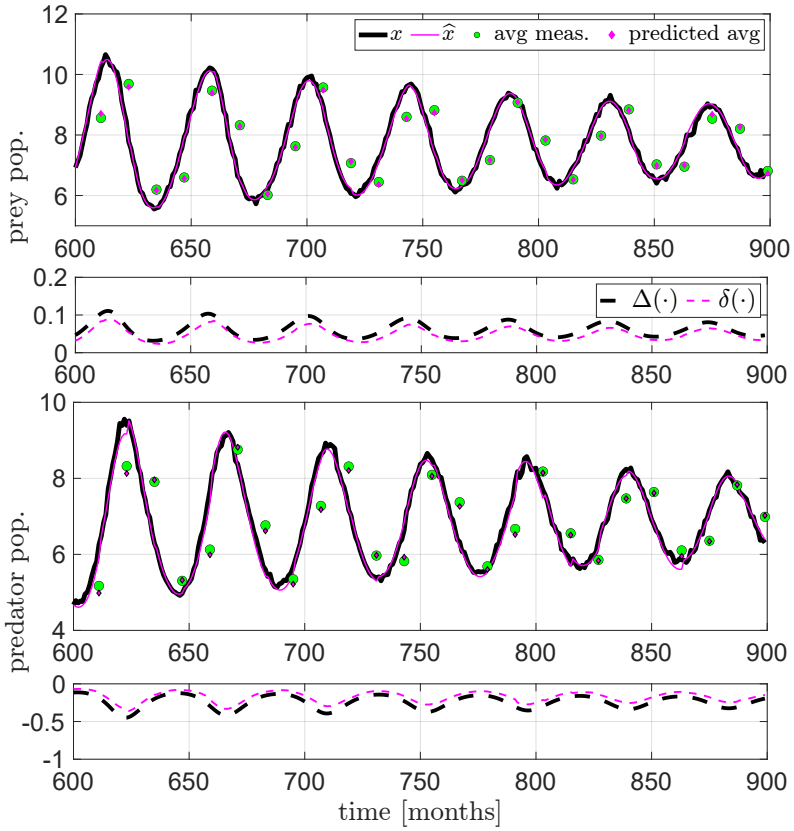


Figure 4.11: True population evolution (black lines) with the averaged measurements (green markers) and predicted population evolution (purple lines) with the reconstructed averages (purple markers) using the model identified from averaged measurements on the validation data. Black and purple dashed lines represent $\Delta(\cdot)$ and $\delta(\cdot)$, respectively.

Discussion and concluding remarks

This chapter has extended the proposed identification framework to the case of non-uniform observations, encompassing scenarios with missing measurements, multiple experimental runs, and temporally aggregated data. Overall, this chapter highlights that handling modeling and observational uncertainties is not a secondary feature but a structural requirement for trustworthy system identification. The developed formulations generalize the identification problem to settings that more faithfully reflect real-world experimental and industrial conditions, where data availability and sampling regularity cannot be guaranteed. Theoretical bounds on the estimation error under missing or aggregated observations have been established, and practical case studies demonstrated that the proposed approach preserves consistency and interpretability even when confronted with incomplete or heterogeneous datasets.

The importance of these results lies in their contribution to bridging the persistent gap between idealized identification assumptions and practical data-collection processes. By formally accounting for non-uniform measurement structures, the framework mitigates the risks of biased parameter estimation that typically arise when simplified models are identified from imperfect data. This provides a principled way to extract reliable physical parameters even in non-ideal conditions, a fundamental capability in modern applications strengthening the use of the identified models in simulation and control.

The following chapter introduces a kernel-based extension of the framework, which removes the need for predefined basis functions and further enhances modeling flexibility while preserving the interpretability of the physical parameters.

Chapter 5

A kernel-based approach to physics-informed identification

This chapter advances the identification framework developed earlier in the thesis by integrating kernel methods with physics-based models. Building on the framework introduced in Chapter 2, where the model of interest was formalized as the nominal physical model plus a sparse selection from a dictionary of basis functions, we now replace ad-hoc basis expansions with a principled, data-based correction learned in a reproducing kernel Hilbert space (RKHS). This preserves interpretability of the physical parameters while systematically compensating for unmodeled dynamics without requiring the definition of a specific dictionary of basis functions.

We begin in Section 5.1 with a static input-output setting and cast the joint estimation of physical parameters and the unknown correction as a regularized optimization problem. An extension of the representer theorem shows that the optimal correction is a finite kernel expansion centered at the observed data, enabling seamless fusion of prior physics with nonparametric flexibility. Section 5.2 extends this construction to state-space models, where not all states are measured. Then, Section 5.3 illustrates the method on two case studies.

Throughout the chapter, we connect to the design principles set in Chapter 2, namely, exploiting partial physics and retaining parameter interpretability, now enhanced by kernel-based approximation in place of pre-defined bases. Readers seeking notation and foundational background on positive-definite kernels, RKHSs, and the reproducing property that underpins the theorems in this Chapter will find a concise recap in Appendix C.

5.1 Kernel-based model integration

Let us start the analysis by considering a nonlinear map of the form

$$y = f(x, \theta) + \Delta(x) + e, \tag{5.1}$$

where $x \in \mathscr{X}$, $y \in \mathbb{R}$ are the input and output, respectively, and $e \in \mathbb{R}$ is an error term which represents measurement noise. Here, $f : \mathscr{X} \times \Theta \to \mathbb{R}$, parametrized in $\theta \in \Theta \subseteq \mathbb{R}^{n_\theta}$, is a known function derived, e.g., from physical principles, whereas $\Delta : \mathscr{X} \to \mathbb{R}$ is an unknown term representing, e.g., modeling errors, uncertainties, or dynamic perturbations. Let a set of T input-output data $\mathscr{D} = \{(x_1, y_1), \dots, (x_T, y_T)\}$ be given, collected from a realization of (5.1) with true parameters $\bar{\theta} \in \Theta$. The goal is to find an estimate $\theta^\star$ of $\bar{\theta}$, and a black-box approximation $\delta(x) : \mathscr{X} \to \mathbb{R}$ of $\Delta(x)$. Such estimate and approximation can be found by solving the following optimization problem:

$$(\theta^\star, \delta^\star) = \arg \min_{\theta \in \Theta, \delta \in \mathscr{H}} \sum_{t=1}^{T} \left[y_t - \hat{y}_t(\theta, \delta) \right]^2 + \gamma \|\delta\|^2_{\mathscr{H}}, \tag{5.2}$$

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.

Before proceeding, we provide two remarks to highlight the differences with the formulation in Chapter 2 and to clarify the role of the regularization weight γ in the kernel-based framework.

Remark 5.1 (Comparison with Chapter 2)**.** *It is worth noting that the cost function adopted in this chapter closely follows the structure introduced in Chapter 2, sharing the same objective of achieving a regularized and physically consistent identification. The key difference lies in the representation of the correction term. In Chapter 2, sparsity is enforced on a finite set of pre-selected basis functions, while here the kernel formulation leverages the representer theorem to construct the correction term directly from the available data. This change eliminates the need for manual basis selection and naturally embeds regularization through the kernel-induced norm. Despite this structural difference, the rationale is the same: to limit the contribution of the black-box component, ensuring that the physical model remains dominant and interpretable, while the nonparametric correction captures only the truly unmodeled dynamics.*

Remark 5.2 (On the role of γ)**.** *The role of γ in* (5.2) *slightly differs from the role it classicaly assumes in standard kernel regularization problems (see, e.g, the one in* (C.2)*). While in standard kernel regularization problems γ balances data fit and regularization, in* (5.2) *it also controls the relative importance assigned to δ and to the physical model f. Specifically, γ determines the trade-off between enforcing the structure provided by the physical model and allowing deviations captured by δ. A smaller γ increases the influence of δ, allowing more flexibility in capturing deviations from the physical model, whereas a larger γ enforces stronger adherence to the model structure. This balance, inherent to any regularization-based method (see, e.g., [96]), 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 chapter.*

The optimization problem in (5.2) aims to estimate the vector of physical parameters associated with the known component of the model f while simultaneously identifying

a function δ that captures the unmodeled term Δ. This approach integrates available prior knowledge, allows the identification of interpretable parameters, and systematically compensates for unmodeled effects, ensuring a more comprehensive and structured representation of the system. Assuming that the unknown term Δ belongs to the RKHS $\mathscr{H}$ associated with the chosen kernel indicates that the solution $\delta^\star$ will admit a kernel representation. Moreover, it also implies that Δ can be effectively approximated using a finite number of kernel evaluations parametrized by the observed data points (see Definition C.2). This assumption is common in nonparametric regression [97] and provides a well-posed framework for learning unmodeled dynamics while ensuring regularization and generalization properties. Moreover, although restricting Δ to a reproducing space, the RKHSs are flexible enough to approximate a broad class of nonlinear functions [98], 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. The kernel-based representation of Δ 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 5.1 (Kernel-based model integration)

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

$$\Gamma(\theta) \doteq [f(x_1, \theta), \dots, f(x_T, \theta)]^\top, \tag{5.3a}$$

$$\omega(\theta) = (\mathbf{K} + \gamma \mathbf{I}_T)^{-1}(Y - \Gamma(\theta)), \tag{5.3b}$$

where $\mathbf{K}$ is the kernel matrix associated to κ and $\mathscr{D}$, having $\mathbf{K}_{ij} = \kappa(x_i, x_j)$, and $\mathbf{I}_T$ denotes the identity matrix of size T. Then, Problem (5.2) admits a solution $(\theta^\star,\ \delta^\star)$ of the form

$$\theta^\star = \arg\min_{\theta \in \Theta} \sum_{t=1}^{T} \Big(y_t - f(x_t, \theta) - \mathbf{K}_t^\top \omega(\theta) \Big)^2 + \gamma \omega(\theta)^\top \mathbf{K} \omega(\theta), \tag{5.4a}$$

$$\delta^\star(x) = \sum_{j=1}^{T} \omega_j^\star \kappa(x, x_j),\ \omega_j^\star \doteq \omega_j(\theta^\star). \tag{5.4b}$$

with $\mathbf{K}_t^\top$ the t-th row of $\mathbf{K}$, and $\omega_j(\theta)$ the j-th element of $\omega(\theta)$.

Proof. Given (5.2) and $\hat{y}_t = f(x_t, \theta) + \delta(x_t)$, define

$$J(\theta, \delta) = \sum_{t=1}^{T} \left[y_t - f(x_t, \theta) - \delta(x_t) \right]^2 + \gamma \|\delta\|_{\mathscr{H}},$$

such that (5.2) can be written as

$$(\theta^\star, \delta^\star) = \arg \min_{\theta \in \Theta, \delta \in \mathscr{H}} J(\theta, \delta).$$

Considering that $\min_{\theta \in \Theta, \delta \in \mathscr{H}} J(\theta, \delta) = \min_{\theta \in \Theta} \min_{\delta \in \mathscr{H}} J(\theta, \delta)$, we have that a minimizer to (5.2) must satisfy

$$\delta^\star(\cdot) = (\arg \min_{\delta \in \mathscr{H}} J(\theta, \delta))|_{\theta = \theta^\star}, \tag{5.5a}$$

$$\theta^\star = \arg \min_{\theta \in \Theta} p(\theta), \tag{5.5b}$$

with $p(\theta) \doteq \min_{\delta \in \mathscr{H}} J(\theta, \delta)$. Here, the inner minimization problem is solved with respect to δ and it now represents a standard kernel regression problem (C.2). Indeed, considering $\tilde{y}_t \doteq y_t - f(x_t, \theta)$, we have

$$\delta^\star(\cdot, \theta) = \arg \min_{\delta \in \mathscr{H}} \sum_{t=1}^{T} \left[\tilde{y}_t - \delta(x_t) \right]^2 + \gamma \|\delta\|_{\mathscr{H}}^2. \tag{5.6}$$

By the representer theorem [67], the optimal solution to (5.6) is

$$\delta^\star(x, \theta) = \sum_{j=1}^{T} \omega_j(\theta) \kappa(x, x_j). \tag{5.7}$$

Considering (5.7) and solving (5.6) as in [99] 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, (5.3b) is obtained given (5.3a) and substituting $\tilde{y}_t \doteq y_t - f(x_t, \theta)$ in $\tilde{Y}$. Moreover, considering the function $p(\theta)$ in (5.5b), we have $p(\theta) = \min_{\delta \in \mathscr{H}} J(\theta, \delta) = J(\theta, \delta^\star(\cdot, \theta))$, which, substituting (5.7), simplifies to

$$p(\theta) = \textstyle\sum_{t=1}^{T} (y_t - f(x_t, \theta) - \mathbf{K}_t^\top \omega(\theta))^2 + \gamma \omega(\theta)^\top \mathbf{K} \omega(\theta), \tag{5.8}$$

noting that

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

from linearity of the inner product and the reproducing property in Definition C.2.b. Thus, we obtain (5.4) by substituting the solution to (5.5b), with $p(\theta)$ given by (5.8), into (5.7), which concludes the proof. □

Theorem 5.1 establishes that the optimal solution to the estimation problem (5.2) can be formulated using kernel-based functions. In particular, the unmodeled component Δ is approximated by δ, defined as a linear combination of kernel evaluations parametrized by the observed data points, thus leading to the following predictive model, representing the optimal solution to Problem (5.2):

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

Moreover, Theorem 5.1 is particularly relevant as it allows for seamless integration of prior physical knowledge with the adaptability of kernel methods, avoiding the use of heuristically chosen basis functions.

Remark 5.3 (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, 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.*

5.1.1 Affine-in-parameters models

A relevant special case arises when the function $f(x, \theta)$ is affine in θ. In this setting, the map (5.1) becomes

$$y = f_0(x) + f(x)^\top \bar{\theta} + \Delta(x) + e, \tag{5.9}$$

where $f_0 : \mathscr{X} \to \mathbb{R}$, $f : \mathscr{X} \to \mathbb{R}^{n_\theta}$, $\bar{\theta} \in \Theta \in \mathbb{R}^{n_\theta}$. Thus, the optimization problem (5.4a) simplifies significantly, leading to a convex optimization problem with a closed-form solution, as formalized in the following theorem.

Theorem 5.2 (Closed-form solution of (5.4))

Consider the same setup of Theorem 5.1. 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 (5.1) is affine in θ, as in (5.9), then the solution of (5.4a) is given by

$$\theta^\star = \left(F(x)^\top \Psi F(x)\right)^{-1} F(x)^\top \Psi Y_0, \tag{5.10}$$

with

$$\Psi \doteq (\mathbf{K} + \gamma \mathbf{I}_T)^{-1}, \tag{5.11}$$

where $\mathbf{K}$ is the kernel matrix associated to κ and $\mathscr{D}$, and γ is the weight controlling the regularization of $\delta(\cdot)$ (5.4b).

Proof. Considering (5.4a) applied to (5.9), we obtain

$$\theta^\star = \arg\min_{\theta\in\Theta} \sum_{t=1}^{T} \Big(y_t - f_0(x_t) - f(x_t)^\top \theta - \mathbf{K}_t^\top \omega(\theta) \Big)^2 + \gamma \omega(\theta)^\top \mathbf{K} \omega(\theta). \tag{5.12}$$

Consider the centered output $y_t - f_0(x_t)$. Rewriting the summation and substituting (5.3) and (5.11), we obtain

$$\theta^\star = \arg\min_{\theta\in\Theta} \|Y_0 - F(x)\theta - \mathbf{K}\Psi(Y_0 - F(x)\theta)\|_2^2 + \gamma (Y_0 - F(x)\theta)^\top \Psi^\top \mathbf{K} \Psi (Y_0 - F(x)\theta) \tag{5.13}$$

where, we used the fact that (5.3a) corresponds to $\Gamma(\theta) = F(x)\theta$, and, according to (5.3b) and (5.11), $\omega(\theta) = (\mathbf{K} + \gamma \mathbf{I}_T)^{-1}(Y_0 - F(x)\theta) = \Psi(Y_0 - F(x)\theta)$. To simplify this expression further, consider

$$\Psi_1 \doteq \mathbf{I}_T - \mathbf{K}\Psi, \tag{5.14a}$$

$$\Psi_2 \doteq \gamma \Psi^\top \mathbf{K} \Psi. \tag{5.14b}$$

Substituting these definitions into (5.13), we write

$$\theta^\star = \arg\min_{\theta\in\Theta} \|\Psi_1(Y_0 - F(x)\theta)\|_2^2 + (Y_0 - F(x)\theta)^\top \Psi_2 (Y_0 - F(x)\theta). \tag{5.15}$$

Problem (5.15) can be recognized as a standard weighted least-squares problem, since the objective is quadratic in $Y_0 - F(x)\theta$, and can be written compactly as

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

Therefore, the optimal solution has the closed form

$$\theta^\star = \big[F(x)^\top (\Psi_1^\top \Psi_1 + \Psi_2) F(x)\big]^{-1} F(x)^\top (\Psi_1^\top \Psi_1 + \Psi_2) Y_0. \tag{5.16}$$

To further simplify this expression, recall from the definition of Ψ in (5.11) 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 (5.14a), we obtain

$$\Psi_1 = \mathbf{I}_T - \mathbf{K}\Psi = \gamma \Psi.$$

Substituting into $\Psi_1^\top \Psi_1 + \Psi_2$ yields

$$\begin{aligned}\Psi_1^\top \Psi_1 + \Psi_2 &= \gamma^2 \Psi^\top \Psi + \gamma \Psi^\top \mathbf{K} \Psi \\ &= \gamma \Psi^\top (\gamma \mathbf{I}_T + \mathbf{K}) \Psi,\end{aligned}$$

factoring out $\gamma \Psi^\top$ and Ψ. Hence, being $(\mathbf{K} + \gamma \mathbf{I}_T)\Psi = \mathbf{I}_T$ from (5.11), and Ψ symmetric, we conclude

$$\Psi_1^\top \Psi_1 + \Psi_2 = \gamma \Psi. \tag{5.17}$$

Substituting (5.17) into (5.16) directly yields (5.10), with the scalar factor γ canceling out as it appears both inside the inverse and outside. This concludes the proof. □

Remark 5.4 (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 C.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 5.2. 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* (5.4a) *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, the optimization problem (5.4a) becomes convex, ensuring a unique and efficiently computable solution. Indeed, the computation of the closed-form solution in (5.10) requires only standard matrix inversions. On the other hand, in the non-affine case, computing the solution to (5.4a) requires iteratively computing the gradient and evaluating the kernel, thus leading to higher computational cost.

Importantly, the structure in (5.9) is quite general, as it does not impose linearity with respect to the input 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 5.2 broadly applicable. On the other hand, when affinity in the parameters does not hold, we can still tackle problem (5.4a) by means of nonlinear programming methods, such as gradient-based techniques, Gauss-Newton-type algorithms, or similar iterative approaches, acknowledging however that due to the potential non-convexity of the problem, the obtained solution may be local.

5.2 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 $f(\cdot)$ and approximating an unknown function $\Delta(x)$ based on measured data pairs (x_t, y_t). However, many physical systems are better described by state-space models, which explicitly capture system dynamics over time [66]. Furthermore, this formulation unifies various prediction models, including nonlinear output error, ARMAX, and ARX models [68], which makes it particularly valuable for system identification. Additionally, many controller and observer design approaches are based on a state-space representation. Unlike static regression models, a state-space formulation accounts for the evolution of hidden states, requiring estimating both system parameters and unmeasured state trajectories.

We thus consider a discrete-time system of the form

$$x_{t+1} = f(x_t, u_t, \bar{\theta}) + \Delta(x_t, u_t) + v_t, \tag{5.18}$$

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 $v_t \in \mathbb{R}^n$ represents the process noise. The functions f and Δ 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 5.1 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 5.1. However, this approach becomes infeasible when certain state components are not directly measurable. In such cases, the system (5.18) is extended to incorporate also the output equation, i.e.,

$$\begin{aligned} x_{t+1} &= f(x_t, u_t, \bar{\theta}) + \Delta(x_t, u_t) + v_t, \\ y_t &= g(x_t, u_t, \bar{\theta}) + w_t, \end{aligned} \tag{5.19}$$

where w_t represents the measurement noise, and the state x_t is not directly accessible.

One common approach to this challenge is multi-step identification, where unmeasured states are recursively estimated through repeated model evaluation (see, e.g., Chapter 2 and [30], [31]). 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 multi-step settings introduces further complexity, as the recursive dependence of δ within f, and consequently θ, prevents a straightforward application of Theorem 5.1. To circumvent these issues, in this section, we adopt an alternative strategy inspired by [100], [101], 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) [102] with an unscented Rauch–Tung–Striebel smoother (URTSS) [103], [104] to reconstruct the hidden state trajectories, enabling kernel-based model integration in a state-space setting.

5.2.1 Nonlinear state reconstruction

We consider the system described by (5.19), where the state variable $x_t \in \mathcal{X}$ evolves according to known 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 $x_{0:T-1} \doteq \{x_0, ..., x_{T-1}\}$ from a given dataset of measurements $\mathscr{D} = \{(u_0, y_0), \dots, (u_{T-1}, y_{T-1})\} \cup \{y_T\}$ and a nominal nonlinear model [103].

First, let us consider the known components of (5.19), which define the nominal model, i.e.,

$$\begin{aligned} x_{t+1} &= f(x_t, u_t, \theta_0) + v_t, \\ y_t &= g(x_t, u_t, \theta_0) + w_t, \end{aligned} \tag{5.20}$$

where θ_0 represents an 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., [69]).

To perform the nonlinear state reconstruction, we propose a nonlinear state smoothing based on a two-step strategy. First, we apply a *forward filtering*, based on an unscented Kalman filter, 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 5.1 (Observability and identifiability)**.** *The system state x is observable along the trajectory induced by the applied input, and the physical parameters θ are identifiable [82] (see, e.g., Assumption 4.1) 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, v_t and w_t, can be characterized as zero-mean Gaussian with covariances $\boldsymbol{P}_v$ and $\boldsymbol{P}_w$, respectively, i.e., $v_t \sim \mathcal{N}(0, \boldsymbol{P}_v)$, $w_t \sim \mathcal{N}(0, \boldsymbol{P}_w)$. The state estimate uncertainty, represented by $\boldsymbol{P}_t$, is initialized as $\boldsymbol{P}_0$ and updated recursively. Therefore, given the state estimate $\hat{x}_{t-1}$ and the covariance

matrix P_{t-1}, we define a matrix of $2n+1$ sigma vectors $\mathbf{X}_{t-1} = [X_{t-1}^{(0)}, \dots, X_{t-1}^{(2n)}]$, with the corresponding weights w_i^m, w_i^c, as

$$\begin{aligned} X_{t-1}^{(0)} &= \hat{x}_{t-1}, \\ X_{t-1}^{(i)} &= \hat{x}_{t-1} + a(\sqrt{P_{t-1}})_i, \qquad i = 1, \dots, n, \\ X_{t-1}^{(i)} &= \hat{x}_{t-1} - a(\sqrt{P_{t-1}})_{i-n}, \quad i = n+1, \dots, 2n, \end{aligned} \tag{5.21}$$

with $(\sqrt{P_{t-1}})_i$ the i-th column of the P_{t-1} matrix square root, and a providing an overall scaling, determined by user-defined parameters regulating the spread of sigma points around the mean and incorporating prior knowledge of the state distribution. Then, for $i = 0, \dots, 2n$, each sigma point propagates through the system dynamics as $X_{t|t-1}^{(i)} = f(X_{t-1}^{(i)}, u_{t-1}, \theta_0)$, so that the preliminary predicted states and covariance are given by

$$\begin{aligned} \hat{x}_t^- &= \textstyle\sum_{i=0}^{2n} w_i^m X_{t|t-1}^{(i)}, \\ P_t^- &= \textstyle\sum_{i=0}^{2n} w_i^c \left(X_{t|t-1}^{(i)} - \hat{x}_t^- \right) \left(X_{t|t-1}^{(i)} - \hat{x}_t^- \right)^T + P_v, \end{aligned} \tag{5.22}$$

with P_v the process noise covariance. Now, we transform the sigma points through the measurement function $Y_{t|t-1}^{(i)} = g(X_{t|t-1}^{(i)})$ and, consequently, we compute the predicted measurements and associated covariance as

$$\begin{aligned} \hat{y}_t &= \textstyle\sum_{i=0}^{2n} w_i^m Y_{t|t-1}^{(i)}, \\ P_{y_t} &= \textstyle\sum_{i=0}^{2n} w_i^c (Y_{t|t-1}^{(i)} - \hat{y}_t)(Y_{t|t-1}^{(i)} - \hat{y}_t)^T + P_w, \end{aligned} \tag{5.23}$$

with P_w the measurement noise covariance. Then, computing $P_{x_t y_t} = \sum_{i=0}^{2n} w_i^c (X_{t|t-1}^{(i)} - \hat{x}_t^-)(Y_{t|t-1}^{(i)} - \hat{y}_t)^T$ and defining the matrix $\mathscr{K}$ as $\mathscr{K} = P_{x_t y_t} P_{y_t}^{-1}$, the filter predictions $\hat{x}_t$, P_t are given by

$$\begin{aligned} \hat{x}_t &= \hat{x}_t^- + \mathscr{K} \left(y_t - \hat{y}_t \right), \\ P_t &= P_{t-1}^- - \mathscr{K} P_{y_t} \mathscr{K}^T. \end{aligned} \tag{5.24}$$

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. In some cases, for instance, the state is augmented to incorporate process and measurement noise. The reader is referred to, e.g., [102], [105], [106] for additional 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, which is described next.

2. Backward smoothing

The backward smoothing phase is based on the unscented Rauch–Tung–Striebel smoother [103], [104] and it aims to obtain the final state estimates. First, given the estimates $\hat{x}_t$, P_t (5.24), for all $t \in [1, T]$ we define the new sigma points $X_t^{(i)}$, $i = 0, \dots, 2n$, using (5.21), and we propagate them as $X_{t+1|t}^{(i)} = f(X_t^{(i)}, u_t, \theta_0)$. Then, we compute the predicted mean, covariance and cross-covariance as

$$\begin{aligned} \hat{x}_{t+1}^- &= \textstyle\sum_{i=0}^{2n} w_i^m X_{t+1|t}^{(i)}, \\ P_{t+1}^- &= \textstyle\sum_{i=0}^{2n} w_i^c (X_{t+1|t}^{(i)} - x_{t+1}^- x)(X_{t+1|t}^{(i)} - x_{t+1}^-)^T + P_v, \\ C_{t+1} &= \textstyle\sum_{i=0}^{2n} w_i^c (X_k^{(i)} - \hat{x}_t)(X_{t+1|t}^{(i)} - \hat{x}_{t+1}^-)^T. \end{aligned} \tag{5.25}$$

Once the smoother gain matrix is defined as $S_t = C_{t+1}(P_{t+1}^-)^{-1}$, the smoothed states and covariances can be computed iterating backwards from $t = T - 1$ to $t = 0$ as

$$\begin{aligned} \hat{x}_t^s &= \hat{x}_t + S_t(x_{t+1}^s - \hat{x}_{t+1}^-), \\ P_t^s &= P_t - S_t(P_{t+1}^- - P_{t+1}^s)S_t^\top, \end{aligned} \tag{5.26}$$

with $\hat{x}_T^s = \hat{x}_T$ and $P_T^s = P_T$, so that we obtain the smoothed state sequence $\hat{x}_{0:T-1}^s \doteq \{\hat{x}_0^s, \dots, \hat{x}_{T-1}^s\}$ and covariances $P_{0:T-1}^s \doteq \{P_0^s, \dots, P_{T-1}^s\}$. Note that the URTSS easily integrates with the UKF as all the necessary data are stored during the filtering step, allowing for efficient utilization in the smoothing process.

5.2.2 State-space reformulation

Once the sequence of unmeasured states has been reconstructed through state smoothing, i.e., $\hat{x}_{0:T-1}^s$, the problem effectively reduces to the one in (5.2), to which Theorem 5.1 is directly applicable. This can be done, for instance, by defining $z_t \doteq [\hat{x}_{t-1}^s, u_{t-1}, u_t]$ as the *new* input variable, and computing the predictions at time t, i.e., $\hat{y}_t(\theta, \delta)$ in (5.2), using the following prediction model

$$\hat{y}_t = \xi(z_t, \theta) + \delta(z_t), \tag{5.27}$$

with $\xi(z_t) \doteq g\big(f(\hat{x}_{t-1}^s, u_{t-1}, \theta), u_t, \theta\big)$. In this formulation, δ captures the discrepancies arising from unknown or unmodeled components in f (5.19), which influence the system evolution and are subsequently mapped to the output space by g.

Remark 5.5 (Estimation without future data)**.** *When employing the identified model for simulation, where both the learned kernel component and the identified parameters are used, future data is not always available to apply the smoother. In this case, the natural solution is to rely solely on the filtering step to provide state estimates up to time t. Beyond this point, the nominal model is used in open-loop to propagate the unmeasured states, which are then fed into the kernel-extended model to generate output predictions.*

Algorithm 3 Kernel-based physics-informed identification

Input: Dataset $\mathscr{D} = \{(u_0, y_0), \dots, (u_{T-1}, y_{T-1})\} \cup \{y_T\}$, f, g, θ_0, $\hat{x}_0$, P_0, P_v, P_w, a, w^m, w^c, κ, γ.
for $t = 1$ to T **do**
    Compute $\hat{x}_t$, P_t using (5.21)–(5.24).
end for
Let $\hat{x}_T^s = \hat{x}_T$.
for $t = T - 1$ to 0 **do**
    From $\hat{x}_t$, P_t, define $\mathbf{X}_t$ using (5.21).
    Compute $X_{t+1|t}^{(i)} = f(X_t^{(i)}, u_t, \theta_0)$, $i = 0, \dots, 2n$.
    Compute $\hat{x}_t^s$ using (5.25), (5.26).
end for
Define new input $z_t \doteq [\hat{x}_{t-1}^s, u_{t-1}, u_t]$.
Define a new prediction model with input z_t, as in (5.27).
Apply Theorem 5.1, and solve (5.4a) with any preferred optimization method, or with Theorem 5.2, if linear in θ.
Output: Estimated parameters $\theta^\star$ and function $\delta^\star$.

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

The overall procedure for implementing the proposed identification approach in a state-space setting is sketched in Algorithm 3.

5.3 Case studies

In this section, we illustrate the effectiveness of the proposed identification method on two case studies. Specifically, we consider an academic example and the cascade tank system (CTS) benchmark [79], also employed in Chapter 2.

5.3.1 Academic example

We consider a regression problem where the goal is to estimate the parameters $\bar{\theta}$ of a linear-in-parameter model, using the kernel-based approach and comparing it to the

solution obtained when no kernel integration is employed, solved with ordinary least-squares methods, the discrepancy modeling approach, and the solution to the standard kernel ridge regression problem (i.e., without embedding prior physical knowledge).

First, 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 the interval $x \in [-1,1]$, employed for training, (ii) 250 samples from $x \in [1,2]$, which are used as a validation set for hyperparameter selection, and (iii) 250 samples from $x \in [-2,-1]$, reserved as test set for performance evaluation. Relying on (5.9), we have

$$\begin{aligned} f_0(x) &= 0,\ f(x)^\top = [1, x, u, x^2, u^2], \\ \Delta(x) &= 0.7\sin(5x) + 0.5\cos(3x) + 0.4x^2 + 0.3x^3 \\ &\quad - 0.2\sin(7x)\cos(2x), \end{aligned} \tag{5.28}$$

where $u = \sin(2\pi x) + 0.5\cos(3\pi x)$ and e follows a Gaussian distribution with standard deviation 0.1. In the selected case study, we select $\bar{\theta} = [2,3,4,1.5,-0.8]$. To improve the estimation accuracy in the presence of unknown nonlinearities $\Delta(x)$, we employ a Laplacian kernel function, where the kernel matrix $\mathbf{K}$ is computed as $\mathbf{K}_{ij} = \exp\left(-\frac{|x_i - x_j|}{\sigma}\right)$.

To select the optimal hyperparameters σ and γ, we rely on a validation-based procedure. Specifically, the training dataset is used to estimate the parameters for different combinations of σ (kernel bandwidth) and γ (regularization weight), while the validation dataset is employed to evaluate the root mean square error (RMSE) of the resulting models. The hyperparameters are tuned over a grid of 50×50 logarithmically spaced values for $\sigma \in \left[10^{-1},10^{1}\right]$, and $\gamma \in \left[10^{-3},10^{1}\right]$, respectively, yielding a total of 2500 candidate pairs. Among all tested combinations, the pair $(\sigma^\star, \gamma^\star) = (0.54, 0.11)$ is selected as it minimizes the RMSE on the validation dataset. The outcome of this procedure is illustrated in Fig. 1, which reports both the validation RMSE (RMSE_{val}) and the parametric error ($\|\bar{\theta} - \theta\|_2$) for each tested configuration of σ and γ. Notice that hyperparameters minimizing the RMSE do not necessarily coincide with those minimizing the parametric error. Indeed, being the true parameter vector $\bar{\theta}$ unknown in practice, it cannot be directly exploited. As shown in Fig. 5.1, the RMSE-based selection nevertheless provides a reliable criterion, yielding parameter estimates that remain close to the minimum parametric error.

To better illustrate the impact of γ on the proposed identification process, we analyze its effect on the validation model performance for fixed $\sigma = \sigma^\star$. Figure 5.2 depicts the RMSE on validation data as a function of γ, highlighting the trade-off between model flexibility and physical consistency. As expected, too small values of γ lead to higher RMSE due to kernel overfitting to deviations and neglecting the physical priors. On the other hand, larger values result in biased parameter estimates as they enforce strict adherence to the physical model at the expense of not capturing relevant unmodeled

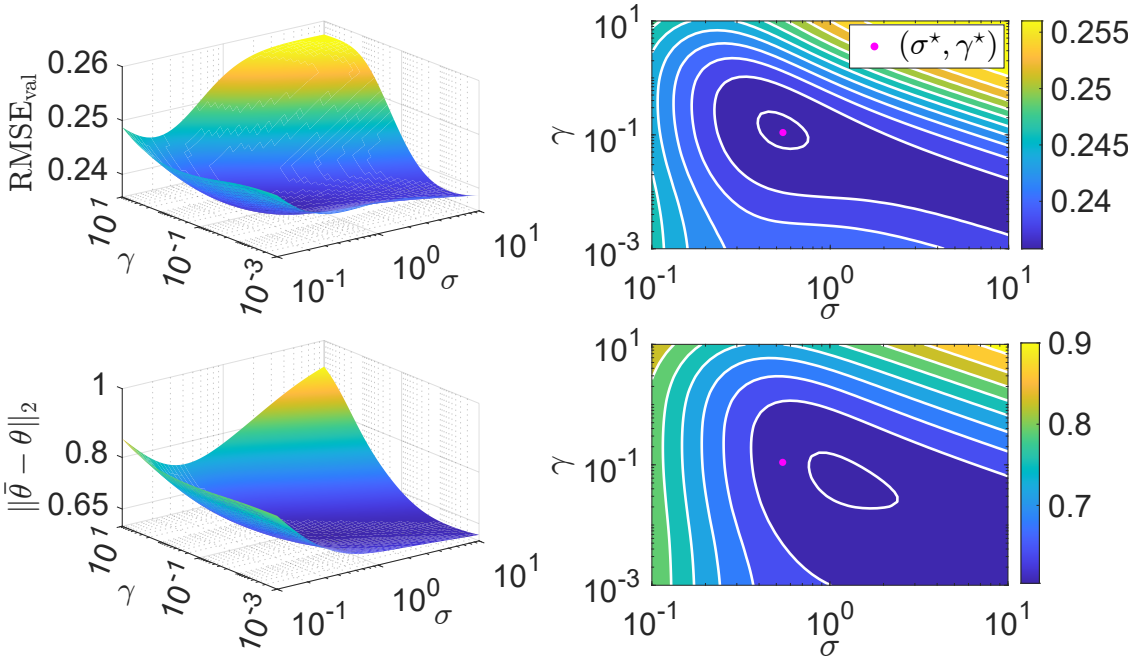


Figure 5.1: Validation RMSE as a function of the kernel bandwidth σ and the regularization weight γ (log scale), with the optimal hyperparameters $(\sigma^\star, \gamma^\star)$ (magenta dot) selected at the minimum RMSE region.

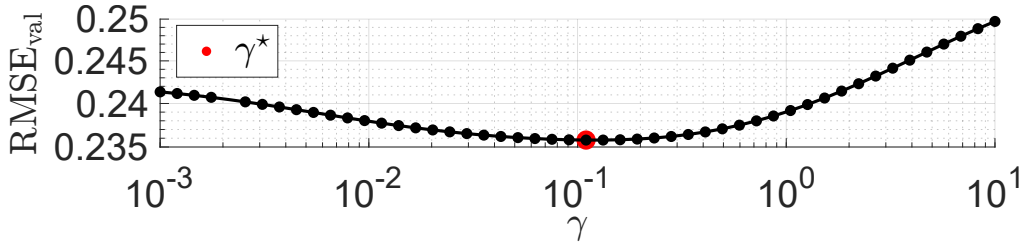


Figure 5.2: RMSE on validation data as a function of γ (log scale).

terms. Clearly, achieving the correct trade-off between physical priors and the kernel-based representation of unmodeled terms is crucial for accurate identification.

Once the hyperparameters are selected, the kernel-based estimator follows the formulation derived in Theorem 5.1 and Theorem 5.2, 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 obtained results are reported in Table 5.1, where the solution obtained with the proposed approach is compared with the true system (to provide easily comparable information on the noise level), the least-squares solution (LS), the discrepancy modeling approach (DM) described in the introduction (see, e.g., [63]), and a straightforward kernel ridge regression (KRR) without the parametric part (i.e., prior knowledge on f_0,

Table 5.1: 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**

f is not exploited)[1].

The estimated parameter vector $\theta^\star$ obtained using (5.10) with $(\sigma^\star, \gamma^\star)$ is significantly closer to the true $\bar{\theta}$ compared to the ordinary least-squares estimate θ^{LS}, which does not account for unknown nonlinearities. Additionally, we calculated the RMSE on the test set (RMSE_{tst}) to assess the accuracy of the predictions. The proposed kernel-based model achieves an RMSE of 0.343, whereas the model obtained using θ^{LS}, which neglects the unmodeled effects, results in a significantly higher RMSE of 0.961.

The estimate θ^{LS} also corresponds to the solution given by the discrepancy modeling approach. In this two-step procedure, the physical parameters are first estimated without accounting for Δ, and the resulting discrepancy is then modeled separately – here, using a Laplacian kernel-based correction. While this approach reduces the RMSE to 0.929, it falls short of the performance achieved by the proposed method, which simultaneously estimates both the physical parameters and the unmodeled dynamics. Moreover, it yields more biased parameter estimates. Notably, the performance of the standard KRR model confirms the importance of physics integration. 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.

Hence, figure 5.3 illustrates a comparison between the estimated function and the measured data for the test and training datasets, confirming that the proposed kernel-based model accurately captures the nonlinear relationship underlying the dataset. These results illustrate the benefits of using a kernel-based integration 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 further assess the robustness and statistical significance of the obtained results, a Monte Carlo analysis is conducted over 1000 independent identification experiments. In each run, a system following the structure in (5.28) is generated with randomly sampled noise realizations and true parameters generated around the nominal values

[1]The same validation procedure was applied to tune the hyperparameters of both the KRR model and the DM approach, yielding $(\sigma^\star = 0.1, \gamma^\star = 10^{-2})$ for DM and $(\sigma^\star = 10, \gamma^\star = 10^{-2})$ for KRR, respectively.

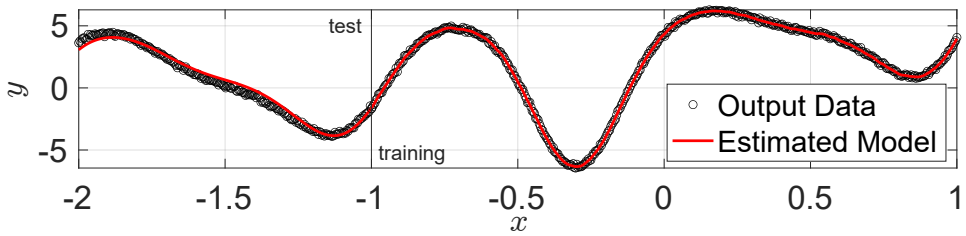


Figure 5.3: Estimated function and measured data for the test and training datasets.

Table 5.2: Statistical evaluation over 1000 Monte Carlo experiments. Mean ± standard deviation are reported.

	$\|\bar{\theta} - \hat{\theta}\|_2$	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	$\mathbf{0.517 \pm 0.056}$	$\mathbf{78.74 \pm 7.67}$	$\mathbf{0.541 \pm 0.138}$

$[2,3,4,1.5,-0.8]$ by adding uniform perturbations up to ±50% of each component's magnitude. The proposed kernel-based method (with Laplacian kernel) is again compared with a standard LS estimator relying only on the known structure, a DM approach obtained by first estimating the LS parameters and then identifying a Laplacian-kernel correction, and a pure kernel ridge regression KRR model without the physical component. For each trial, training, validation, and test sets were defined as before, and hyperparameters for the proposed, DM, and KRR approaches were selected through the same validation-based procedure.

Table 5.2 reports the average parameter estimation error, the fit percentage ($\text{fit} \doteq 100\left(1 - \|y - \hat{y}\|_2 / \|y - \bar{y}\|_2\right)$, 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.

5.3.2 Cascade tank system benchmark

In the second example, we test the proposed approach on the CTS benchmark [79] and compare it with other state-of-the-art estimation methods. As described in Chapter 2, the CTS regulates fluid levels among two connected tanks and a pump. First, water is pumped into the upper tank, then it flows to the lower tank and back to the reservoir. Overflow occurs when a tank is full: The excess from the upper tank partially drains into the lower one, with the rest exiting the system. In [79] an approximate non-linear,

continuous state-space model of the CTS is derived using the Bernoulli's and mass conservation principles. Here, we rely on its discretized version, i.e.,

$$\begin{aligned} x_{1,t+1} &= x_{1,t} + T_s\left(-k_1\sqrt{x_{1,t}} + k_4 u_t + v_{1,t}\right), \\ x_{2,t+1} &= x_{2,t} + T_s\left(k_2\sqrt{x_{1,t}} - k_3\sqrt{x_{2,t}} + v_{2,t}\right), \\ y_t &= x_{2,t} + w_t, \end{aligned} \tag{5.29}$$

where $u_t \in \mathbb{R}$ is the input signal, $x_{1,k} \in \mathbb{R}$ and $x_{2,k} \in \mathbb{R}$ are the states of the system, $y_t \in \mathbb{R}$ is the output, and $e_t \in \mathbb{R}^2$, $w_t \in \mathbb{R}$ are the additive noise sources. The sampling time is set to $T_s = 4s$. Moreover, 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. Since this model ignores the water overflow effect, unmodeled dynamics are included in the physical dynamics model (5.29). The training and validation datasets consist of $T = 1024$ input-output samples. 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, described by Algorithm 3, assuming that f and g are defined according to the discretized model (5.29) and selecting $P_v = 10^{-3} I_2$, $P_w = 10^{-2}$, $P_0 = 0.5 I_2$, $\theta_0 = [0.05, 0.05, 0.05, 0.05]^\top$, and $\hat{x}_0 = [y_0, y_0]^\top$. Moreover, we set the UKF weights according to the formulation in [102], 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 $i = 1, \dots, 2n$. Then, we solve an optimization problem of the form (5.2) for $\gamma = 10^{-1}$. Specifically, once the smoothed trajectory of the unmeasured state $\hat{x}^s_{1,0:T-1}$ is computed, we define a predictive model of the form (5.27) and we solve Problem (5.2) applying Theorem 5.1. Specifically, we employ the *fmincon* solver using a *sqp* method to solve Problem (5.4a). Hence, considering (5.29) and selecting $z_t \doteq [\hat{x}^s_{1,t-1}, y_t, u_{t-1}]$, we define

$$\xi(z_t) \doteq y_t + T_s k_2 \sqrt{\hat{x}^s_{1,t-1} + T_s\left(-k_1\sqrt{\hat{x}^s_{1,t-1}} + k_4 u_{t-1}\right)} - T_s k_3 \sqrt{y_t}.$$

Thus, once we obtain $(\theta^\star, \delta^\star)$ as the solution of (5.2) according to Theorem 5.1, 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, as suggested in [79], the RMSE is employed as the performance metric. Table 5.3 presents the performance of the proposed kernel-based identification method compared with the solution obtained by solving (5.2) when no kernel is used to compensate for unmodeled dynamics in (5.29). 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,

Table 5.3: Performance analysis on estimation and validation data.

	Pred. (RMSE)		**Sim.** (RMSE, fit)	
	Train.	Val.	Train.	Val.
(5.29) **only**	0.06	0.06	0.37, 83.14%	0.37, 82.46%
(5.29) **+ Kernel**	0.04	0.05	0.17, 92,11%	0.18, 91.55%

Table 5.4: Methods comparison (simulation RMSE on validation data).

Method	**RMSE**	**Method**	**RMSE**
Svensson et al. [16]	0.45	PWARX [17]	0.35
Volt.FB [23]	0.39	SED-MPK [24]	0.48
INN [25]	0.41	PNLSS-I [80]	0.45
NLSS2 [80]	0.34	NOMAD [81]	0.37
Chapter 2 [69]	0.26	**Proposed (kernel)**	0.178

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 5.3. These first results highlight the significant reduction in the RMSE achieved through kernel integration, 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 multi-step simulation accuracy. This improvement is also reflected in the results reported in Table 5.4, where we compare the simulation performance for the validation data with other state-of-the-art approaches from the literature. Among these methods, we observe that the proposed approach also outperforms the solution of the multi-step identification approach with basis functions for black-box compensation, presented in Chapter 2. This improvement is likely due to the ability of the kernel-based method to automatically select an optimal representation for the approximating term δ, whereas in Chapter 2 this term was chosen from a limited, predefined dictionary of basis functions.

Finally, we report the identified parameters for completeness. The smoothed initial condition is $\hat{x}_0^s = [4.78, 5.20]$ whereas, for $\hat{k}$, we have: (i) with no kernel, $\hat{k} = [-0.01, 0.05, 0.06, 0.01]$, and (ii) with kernel, $\hat{k} = [0.08, 0.05, 0.05, 0.06]$.

Discussion and concluding remarks

This chapter has introduced a kernel-based extension of the proposed identification framework, providing a principled and flexible approach to capture unmodeled dynamics while preserving the interpretability of the physical parameters. By embedding the model residuals within a reproducing kernel Hilbert space, the framework eliminates

the need for manually defined basis functions and, once the kernel type is selected, allows the data themselves to determine the functional structure of the corrective term. This integration between kernel-based learning and physics-informed modeling leads to a unified formulation that naturally balances flexibility and physical consistency.

From an identification perspective, this represents an important step forward. The kernel-based approach not only improves predictive accuracy but also strengthens the conceptual link between physics-based models and data-driven corrections, showing that model uncertainty can be addressed systematically through reproducing kernel theory. Notably, the integration with state estimation techniques such as unscented Kalman filtering enables reliable multi-step performance even though the cost function is formally one-step. This property, which emerges from the recursive propagation of the filtering scheme, allows the identified models to exhibit consistency and stability over longer horizons, an aspect that is currently under further investigation to establish more rigorous theoretical guarantees. Overall, the resulting formulation yields interpretable, reliable, and robust models even in complex, nonlinear scenarios, where simplified physical descriptions would otherwise fail to capture the full system behavior.

The next and final chapter discusses the broader implications of the proposed framework, highlighting its contributions, limitations, and future directions toward scalable and interpretable physics-informed system identification.

Chapter 6

Discussion and conclusions

This thesis presented a unified framework for the physics-informed identification of nonlinear dynamical systems, bridging model-based and data-driven paradigms through the explicit integration of domain knowledge and sparse learning principles. The proposed approach builds upon three main pillars: multi-step optimization, sparse black-box augmentation, and kernel-based generalization, each contributing to a progressively broader and more flexible identification scheme.

Summary of contributions

The first part of the thesis introduced a *multi-step system identification framework* designed to improve physical interpretability and long-term accuracy of identified models. By minimizing the cumulative prediction error over extended horizons, the proposed method addresses the typical mismatch between single-step training and multi-step deployment, ensuring that the identified dynamics remain accurate over time.

The second contribution proposed a *sparse black-box augmentation* approach that complements a known physical model with a minimal set of corrective terms. This strategy effectively captures unmodeled dynamics without compromising physical interpretability while allowing joint estimation of physical parameters and residual dynamics within a single optimization problem. Rigorous results established sufficient conditions for exact sparsity recovery and bounded parametric estimation error, thereby providing a solid theoretical foundation for physics-informed and sparse identification.

The third contribution extended the framework to the setting of *non-uniform observations*, including multiple experimental runs, missing samples, and temporally aggregated measurements. The developed theory quantified the influence of irregular sampling on the identification performance and demonstrated that the proposed methods maintain robustness and accuracy even under heterogeneous data collection conditions. This extension broadens the practical applicability of the framework to real-world scenarios where ideal measurement setups are rarely available.

Finally, the thesis introduced a *kernel-based extension* that replaces the explicit arbitrarily chosen dictionaries of basis functions with a data-based representation in a reproducing kernel Hilbert space. This formulation removes the need for manual basis functions design, proposing a systematic kernel-based correction of the physical model while preserving interpretability. When combined with state estimation techniques such as Kalman filtering and smoothing, the resulting model achieves improved prediction accuracy and noise robustness in nonlinear state-space systems.

Discussion and outlook

The presented framework provides a systematic approach for integrating physical insight and data-driven flexibility in nonlinear system identification. From a theoretical perspective, it offers guarantees on parameter estimation and sparsity, while maintaining computational tractability through the use of first-order optimization and automatic differentiation, and flexibility to non-uniform data. From a practical standpoint, it demonstrates how mixed modeling can enhance both predictive reliability and interpretability, two properties that are often at odds in modern learning-based methods.

Despite these achievements, several open challenges remain. Although the proposed framework ensures interpretability and scalability through sparsity- and kernel-based regularization, the selection of such regularization parameters still relies on heuristic tuning. Future work will therefore focus on developing systematic, theoretically grounded selection criteria derived, e.g., from Lipschitz analysis, enabling an automatic and principled trade-off between model fidelity and physical consistency. Within this context, recent results (see, e.g., [107]) have shown that enforcing physical properties, interpreted through incremental input-output or integral quadratic constraints, via explicit control of the kernel's Lipschitz constant can enhance the physical consistency of nonlinear models. This perspective suggests that constraining the learning process within a Lipschitz-consistent kernel space not only guarantees well-posedness and robustness of the identified dynamics but also improves accuracy when models are deployed in feedback or interconnected settings. Embedding this Lipschitz-centered principle directly into the proposed identification framework would thus enable the construction of models that are interpretable, accurate and provably well-behaved. Furthermore, ongoing developments open a promising avenue toward certified multi-step prediction error bounds. By explicitly designing and regulating the Lipschitz constant during the identification process, future research will aim to derive models that guarantee bounded long-horizon simulation errors by construction, thereby establishing a formal bridge between kernel regularization, robust learning, and physics-informed identification. In this perspective, *Lipschitz-based identification* emerges as a unifying framework that consolidates these elements into a coherent, theoretically grounded approach to reliable nonlinear modeling.

A second natural evolution of this research lies in extending the proposed methodology to interconnected and distributed dynamical systems. In such settings, the identification of both local dynamics and interconnection structures could benefit from the developed multi-step and kernel formulations, providing a foundation for scalable, network-aware physics-informed identification.

Furthermore, while this thesis has primarily focused on identification accuracy and theoretical guarantees in an open-loop setting, a critical direction for future works is the assessment of these models within closed-loop control architectures. The ultimate validation of an identified model often lies in its utility for decision-making, particularly in safety-critical scenarios. Therefore, a natural extension of this work is to perform a rigorous comparative analysis against state-of-the-art data-driven control techniques, such as direct data-driven control and predictive control strategies (see, e.g., [108], [109]). Such a comparison would aim to quantify the specific benefits of the proposed physics-informed framework over purely data-driven approaches. Specifically, future work will investigate whether the preservation of physical priors and the enforcement of structural constraints (as detailed in Chapter 2) translate into superior closed-loop performance.

In conclusion, this thesis establishes a versatile and theoretically grounded foundation for physics-informed identification of nonlinear systems. It demonstrates that the synergy between physical modeling, optimization, and data-based learning principles can yield models that are not only accurate but also interpretable and reliable. The methodologies developed herein are thus expected to serve as building blocks for future advances in mixed modeling, control-oriented learning, and large-scale dynamical system analysis.

Appendix A

Multi-step optimization

This Appendix addresses complementary aspects of the identification framework presented througout the thesis and proposes two additional contributions to the field of system identification. First, drawing inspiration from Neural Network training, it introduces a tool for solving identification problems by leveraging first-order optimization and Automatic Differentiation (AD), which is employed to solve the optimization tasks encountered throughout this thesis. The proposed method exploits gradients with respect to the parameters to be identified and leverages Linear Parameter-Varying (LPV) sensitivity equations to model gradient evolution. Second, it demonstrates that the computational complexity of the proposed method is linear in both the multi-step horizon length and the parameter size, ensuring scalability for large identification problems. The analysis proposed in this Appendix will be complemented in Appendix B, where conditions for a reliable and efficient optimization and identification process for dynamical systems are proposed. The content of this Appendix is based on the published paper [110].

This Appendix is organized as follows. Section A.1 revisits the main aspects of multi-step identification, emphasizing the implications for optimization. Section A.2 discusses the role of automatic differentiation, detailing both the gradient computation in LPV settings and the proposed approach to efficiently propagate derivatives. Finally, Section A.3 analyzes the computational complexity of the methods.

Within this Appendix, the symbol ∇ denotes the gradient operator, where $\nabla_x L$ represents the vector of partial derivatives of $L(x)$ with respect to x. The Jacobian matrix of $v_k \in \mathbb{R}^{n_v}$ with respect to $w_k \in \mathbb{R}^{n_w}$ is denoted as $\mathscr{J}_k^{v/w} \in \mathbb{R}^{n_v, n_w}$, i.e., $\frac{\partial v_k}{\partial w_k}$. Similarly, $\mathscr{J}_k^{v/v} \in \mathbb{R}^{n_v, n_v}$ is the Jacobian matrix of v_k with respect to v_{k-1}, i.e., $\frac{\partial v_k}{\partial v_{k-1}}$.

A.1 Multi-step identification

Multi-step identification involves minimizing a multi-step cost function over a horizon T, propagating the predictions $\hat{x}_k$ over the desired horizon, and recursively applying the

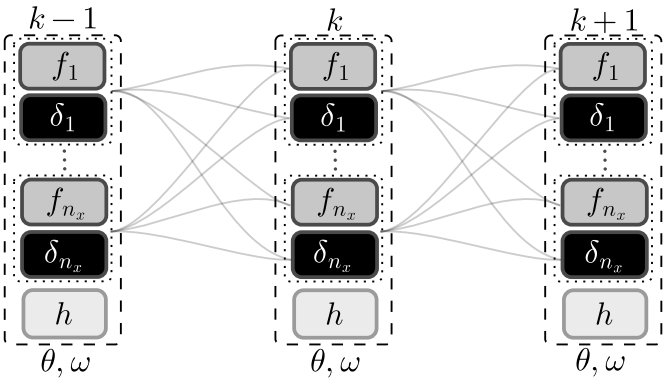


Figure A.1: Multi-step model propagation.

dynamical model $\mathscr{M}$, evaluating the multi-step prediction error e_k. A somewhat similar approach is found in RNNs [26], which aim at accurately estimating an output sequence by minimizing the sum-of-squares error measure over a finite horizon, involving the recursion of the hidden states repeatedly through the same layer. This implies that the network weights, as well as the architectures, are the same over the entire horizon. Thus, recalling the typical structure of neural networks, it is possible to represent the multi-step propagation of a model of the form (2.3) as in Fig. A.1, drawing an analogy between RNN layers and the dynamical model propagation. Notice that this representation is very general, and can be used for *any* dynamical system of the form (2.3). As in RNNs, during the identification process, we need to consider the weights, i.e., parameters to identify, and the structure, i.e., functional form of the model, to be the same at each time step along the prediction horizon T, in order to preserve physical consistency and temporal dependencies in the learning process. This similarity motivates an investigation on the techniques commonly used for optimizing recurrent neural networks, such as backpropagation through time [111], [112], and, more in general, automatic differentiation, in the context of multi-step system identification.

In the next section, we focus on analyzing how the gradient may be efficiently evaluated in multi-step identification problems.

A.2 Automatic differentiation in multi-step identification

At the core of first-order techniques is the computation of the gradient of the cost function $\mathscr{C}_T$ evaluated at the current solution. For instance, in standard gradient descent, once the gradients with respect to the parameter $\nabla_\theta \mathscr{C}_T \in \mathbb{R}^{n_\theta}$, and the initial condition $\nabla_{x_0} \mathscr{C}_T \in \mathbb{R}^{n_x}$, i.e., $\nabla_\theta \mathscr{C}_T = \left[\frac{\mathrm{d}\mathscr{C}_T}{\mathrm{d}\theta_i} \right], \forall i \in \left[1, n_\theta\right]$, $\nabla_{x_0} \mathscr{C}_T = \left[\frac{\mathrm{d}\mathscr{C}_T}{\mathrm{d}x_{0,i}} \right], \forall i \in \left[1, n_x\right]$, are

computed, the weights are updated in the direction that minimizes the total cost, i.e.,

$$\hat{\theta} \leftarrow \hat{\theta} - \varsigma_\theta \nabla_\theta \mathscr{C}_T, \quad \hat{x}_0 \leftarrow \hat{x}_0 - \varsigma_{x_0} \nabla_{x_0} \mathscr{C}_T, \tag{A.1}$$

where ς_θ and ς_{x_0} are the learning rates, that can be designed exploiting state-of-the-art methods [113]. In this context, automatic differentiation is a general computational technique used to efficiently compute the gradients of functions, often applied in optimization and machine learning. It relies on the concept of graph computation, where a function is decomposed into a series of elementary operations represented as nodes, with edges indicating data flow between them. Thus, AD systematically applies the chain rule of calculus to propagate derivatives through this graph. By evaluating both the function and its derivatives simultaneously, AD avoids numerical errors typically associated with finite difference methods and is computationally more efficient than symbolic differentiation [114]. In the context of multi-step identification of dynamical systems, the computational graph becomes very specialized due to the recursive nature of the problem. This recursive structure allows for efficient gradient evaluation at each time step, effectively turning the gradient evolution into a dynamical system itself, directly connected to the estimation model (2.3), exploited to identify the system under analysis, and making the process both general and well-suited for the identification of any dynamical system. In this section, we exploit the explicit knowledge of the structure of the dynamical system (2.1) and the defined estimation model (2.3), and present an efficient adaptation of the forward automatic differentiation technique to the context of system identification.

Remark A.1 (On the gradient LPV equations)**.** *The following formulation has appeared in the literature under various forms and names, such as adjoint sensitivity analysis and state sensitivity equations (see, e.g., [56], [68]). In this thesis, we present a tailored version designed for the identification of nonlinear dynamical systems. This formulation will be exploited in Section B.1 to perform the stability analysis of the gradient, providing insights to ensure its boundedness during the optimization process.*

Clearly, various methods are available in the literature for calculating the gradient, with varying degrees of approximation, and, at least in principle, all of these are applicable to the proposed framework, varying from numerical differentiation to automatic differentiation [114]. In this Appendix, we exploit the explicit knowledge of the function $f(\cdot)$ in (2.1), and propose an efficient method to compute the exact gradient. The method is inspired by the classical backpropagation scheme, adopted, e.g., in neural networks, and recently proposed in the context of system identification in [112], and similarly in [115]. Differently from backpropagation-through-time [111], where the error is back-propagated in time in an unfolded RNN only after forward-propagating the inputs through the unfolded network, we represent the evolution of the gradient by studying the predictions and errors of the system as they evolve forward-in-time. Thus,

we propagate the gradient through the recursion intrinsically defined in dynamical systems updating it while propagating the predicted states. This involves the definition of a "memory" of the effects of past errors and an "innovation" due to current errors.

The solution we propose presents several distinguishing characteristics that make it of particular interest. First, the method produces exact gradient values and offers an efficient computational performance. This is obtained by propagating forward in time the impact of $\hat{\theta}$ and $\hat{x}_0$ on current predictions, exploiting the definition of a memory matrix, and thus avoiding the necessity of multiplying similar matrices as in the backpropagation scheme. Second, it allows the definition of a particular time-varying, discrete-time dynamical system describing the evolution of the gradient with respect to parameters and initial condition. The closed-form formula for the evolution of the gradient not only reduces the computational cost but also allows to study analytically its behavior during the identification process. Specifically, we will show in Section B.1 how it can be used to derive sufficient conditions for avoiding the phenomenon of so-called "exploding gradient".

A.2.1 Gradient LPV equations

The following results are presented considering a cost function of the form

$$\mathscr{C}_T(\theta, x_0, \omega; \mathbf{e}_{0:T-1}, \hat{\mathbf{x}}_{1:T-1}) \doteq \sum_{k=0}^{T-1} \mathscr{L}_k\left(\theta, x_0, \omega; e_k, \hat{x}_k\right), \tag{A.2}$$

where $\mathscr{L}_k$ is intended as a general, twice continuously differentiable loss function, including possible physics-based penalties and regularization terms (see Chapter 2). Specifically, with the following proposition, we show how to represent the evolution of the gradient with respect to θ as a dynamical system.

Proposition A.1 (Gradient dynamics – θ)

Define the *memory matrix* $\Lambda_k \doteq \frac{\mathrm{d}x_k}{\mathrm{d}\theta} \in \mathbb{R}^{n_x, n_\theta}$, as the matrix containing the total derivatives of the states with respect to θ. Define vectors $\rho_k \in \mathbb{R}^{n_x}$, $\varrho_k \in \mathbb{R}^{n_\theta}$ as

$$\rho_k \doteq \left(\nabla_e^\top \mathscr{L}_k \mathscr{J}_k^{e/y} \mathscr{J}_k^{y/x}\right)^\top, \quad \varrho_k \doteq \nabla_\theta \mathscr{L}_k. \tag{A.3}$$

Then, the gradient evolution with respect to the parameter θ over the multi-step horizon T is described by the following time-varying dynamical system

$$\Lambda_k = \mathscr{J}_k^{x/x} \Lambda_{k-1} + \mathscr{J}_k^{x/\theta}, \tag{A.4a}$$

$$\nabla_\theta \mathscr{C}_k = \nabla_\theta \mathscr{C}_{k-1} + \Lambda_k^\top \rho_k + \varrho_k, \tag{A.4b}$$

for $k = 1, \dots, T$, with $\Lambda_0 \doteq \frac{\mathrm{d}x_0}{\mathrm{d}\theta} = \mathbf{0}_{n_x, n_\theta}$, $\nabla_\theta \mathscr{C}_0 = \mathbf{0}_{n_\theta}$.

Proof. Consider (A.2). The gradient with respect to θ can be computed as $\nabla_\theta \mathscr{C}_T \doteq \frac{\mathrm{d}\mathscr{C}_T}{\mathrm{d}\theta} = \frac{\mathrm{d}}{\mathrm{d}\theta}\left(\sum_{k=1}^{T} \mathscr{L}_k\right) = \sum_{k=0}^{T} \frac{\mathrm{d}\mathscr{L}_k}{\mathrm{d}\theta}$. It follows that

$$\nabla_\theta \mathscr{C}_k = \nabla_\theta \mathscr{C}_{k-1} + \frac{\mathrm{d}\mathscr{L}_k}{\mathrm{d}\theta}, \tag{A.5}$$

where $\nabla_\theta \mathscr{C}_k \doteq \sum_{\tau=0}^{k} \frac{\mathrm{d}\mathscr{L}_\tau}{\mathrm{d}\theta}$ and $\nabla_\theta \mathscr{C}_0 = \mathbf{0}_{n_\theta}$. Exploiting the chain rule of differentiation we have the following relation

$$\frac{\mathrm{d}\mathscr{L}_k}{\mathrm{d}\theta}^\top = \frac{\partial \mathscr{L}_k}{\partial \theta}^\top + \frac{\partial \mathscr{L}_k}{\partial e_k}^\top \frac{\partial e_k}{\partial y_k} \frac{\partial y_k}{\partial x_k} \frac{\mathrm{d}x_k}{\mathrm{d}\theta} \tag{A.6}$$

where the last term is defined as

$$\frac{\mathrm{d}x_k}{\mathrm{d}\theta} = \frac{\partial x_k}{\partial \theta} + \frac{\partial x_k}{\partial x_{k-1}} \frac{\mathrm{d}x_{k-1}}{\mathrm{d}\theta}. \tag{A.7}$$

Considering the memory matrix $\Lambda_k = \frac{\mathrm{d}x_k}{\mathrm{d}\theta}$, we can rewrite (A.7) as $\Lambda_k = \frac{\partial x_k}{\partial \theta} + \frac{\partial x_k}{\partial x_{k-1}} \Lambda_{k-1} = \mathscr{J}_k^{x/\theta} + \mathscr{J}_k^{x/x} \Lambda_{k-1}$, which yields (A.4a). Thus, (A.6) can be rewritten as

$$\frac{\mathrm{d}\mathscr{L}_k}{\mathrm{d}\theta}^\top = \nabla_\theta^\top \mathscr{L}_k + \nabla_e^\top \mathscr{L}_k \mathscr{J}_k^{e/y} \mathscr{J}_k^{y/x} \Lambda_k^\top = \varrho_k^\top + \rho_k^\top \Lambda_k, \tag{A.8}$$

using (A.3). Thus, (A.4b) is obtained by transposing and substituting (A.8) in (A.5), concluding the proof. □

In (A.4a) the term $\mathscr{J}_k^{x/\theta}$ reflects the direct effect of the model parameter estimation $\hat{\theta}$ on the current state prediction $\hat{x}_k$ and, consequently, on $\mathscr{L}_k$. Conversely, the term Λ_{k-1} encapsulates how the effect of $\hat{\theta}$ on past predictions, i.e., $\hat{x}_\tau$ with $\tau \in [1, k-1]$, has affected the current state estimation $\hat{x}_k$. Thus, (A.4b) defines a formula in which the gradient is updated at each time step k with an *innovation term*, i.e., $\Lambda_k^\top \rho_k + \varrho_k$, exploiting the information encapsulated in Λ_k. Here, notice that $\mathscr{J}_k^{e/y}, \mathscr{J}_k^{y/x}, \mathscr{J}_k^{x/x}, \mathscr{J}_k^{x/\theta}$ are time-varying matrices with fixed structure and shape, only depending on the values of $\hat{x}_k, \tilde{u}_k, \hat{y}_k, \tilde{y}_k, e_k, \hat{\theta}, \hat{x}_0$, and ω at time-instant k. For instance, we have that

$$\mathscr{J}_k^{x/x} = \frac{\partial \hat{x}_k}{\partial \hat{x}_{k-1}} = \frac{\partial \left(f(\hat{x}_{k-1}, \tilde{u}_{k-1}; \hat{\theta}) + \delta\left(\hat{x}_{k-1}, \tilde{u}_{k-1}; \omega\right)\right)}{\partial \hat{x}_{k-1}}.$$

While an analogous result can be obtained considering the extended parameter vector $\vartheta = [\theta^\top, \omega^\top]^\top \in \mathbb{R}^{n_\theta + n_\omega}$, a slightly different dynamic is obtained in the case of the gradient with respect to the initial condition. This is shown in the following proposition.

Proposition A.2 (Gradient dynamics – x_0)

Let $\Lambda_{0,k} \doteq \frac{\mathrm{d}x_k}{\mathrm{d}x_0} \in \mathbb{R}^{n_x,n_x}$ be the matrix containing the total derivatives of the states with respect to x_0. Consider (A.3). The gradient evolution with respect to the initial condition along the multi-step horizon $\boldsymbol{T}$ is obtained by means of the following time-varying dynamical system

$$\Lambda_{0,k} = \mathscr{J}_k^{x/x} \Lambda_{0,k-1}, \tag{A.9a}$$

$$\nabla_{x_0} \mathscr{C}_k = \nabla_{x_0} \mathscr{C}_{k-1} + \Lambda_{0,k}^\top \rho_k, \tag{A.9b}$$

with $\Lambda_{0,0} \doteq \frac{\mathrm{d}x_0}{\mathrm{d}x_0} = \mathbf{I}_{n_x}$, $\nabla_{x_0} \mathscr{C}_0 = \Lambda_{0,0}^\top \rho_0$.

Proof. Let us consider

$$\nabla_{x_0} \mathscr{C}_k = \nabla_{x_0} \mathscr{C}_{k-1} + \frac{\mathrm{d}\mathscr{L}_k}{\mathrm{d}x_0}. \tag{A.10}$$

Exploiting the chain rule we have

$$\frac{\mathrm{d}\mathscr{L}_k}{\mathrm{d}x_0}^\top = \frac{\partial \mathscr{L}_k}{\partial e_k}^\top \frac{\partial e_k}{\partial y_k} \frac{\partial y_k}{\partial x_k} \frac{\mathrm{d}x_k}{\mathrm{d}x_0}, \tag{A.11}$$

where the last term is defined as $\frac{\mathrm{d}x_k}{\mathrm{d}x_0} = \frac{\partial x_k}{\partial x_{k-1}} \frac{\mathrm{d}x_{k-1}}{\mathrm{d}x_0}$. Thus, recalling $\Lambda_{0,k} = \frac{\mathrm{d}x_k}{\mathrm{d}x_0} \in \mathbb{R}^{n_x,n_x}$, we can write

$$\Lambda_{0,k} = \frac{\partial x_k}{\partial x_{k-1}} \Lambda_{0,k-1} = \mathscr{J}_k^{x/x} \Lambda_{0,k-1},$$

which yields (A.9a). Then, (A.11) can be rewritten as

$$\frac{\mathrm{d}\mathscr{L}_k}{\mathrm{d}x_0}^\top = \nabla_e^\top \mathscr{L}_k \mathscr{J}_k^{e/y} \mathscr{J}_k^{y/x} \Lambda_{0,k} = \rho_k^\top \Lambda_{0,k}, \tag{A.12}$$

using (A.3). Thus, (A.9b) is obtained by transposing and substituting (A.12) in (A.10). □

In this case, we note that there is no "direct" effect of $\hat{x}_0$ on $\hat{x}_k$, since, differently from $\hat{\theta}$, the estimated initial condition does not affect future predictions entering "directly" into the model at each time step, but only through its propagation over time. This effect is captured by the "memory" term Λ_0.

A.2.2 Proposed approach

Building on Propositions A.1 and A.2, Algorithm 4 summarizes the proposed multi-step identification procedure. The estimation model (e.g., (2.3)) is propagated with initial state $\hat{x}_0$, parameters $\hat{\theta}$ and ω, while gradients are updated along the horizon T using the derived dynamics. Parameters are then iteratively updated until either: i) the loss function reaches a value below the threshold ε_1, or ii) the gradient norm becomes smaller than the threshold ε_2.

Algorithm 4 First-order identification algorithm

Given a dataset $\{\tilde{\mathbf{u}}_{0:T-1}, \tilde{\mathbf{y}}_{0:T-1}\}$, collected from (2.1), choose $\delta(\cdot)$, ε_1, and ε_2.
Initialize $\hat{x}_0$, $\hat{\vartheta} = [\hat{\theta}^\top, \omega^\top]^\top$.
while $\mathscr{C}_T \geq \varepsilon_1$ **and** $\|\nabla \mathscr{C}_T\|_2 \geq \varepsilon_2$ **do**
    Initialize $k = 0$, $\Lambda_0 = \mathbf{0}_{n_x, n_\theta}$, $\nabla_\vartheta \mathscr{C}_0 = \mathbf{0}_{n_\vartheta}$, $\Lambda_{0,0} = \mathbf{I}_{n_x}$, and $\nabla_{x_0} \mathscr{C}_0 = \Lambda_{0,0}^\top \rho_0$.
    while $k \leq T - 1$ **do**
        Predict $\hat{x}_{k+1}$, $\hat{y}_k$ using the estimation model (2.3), with $\hat{x}_0$, $\hat{\vartheta}$.
        Compute $e_k = \hat{y}_k - \tilde{y}_k$ and $\mathscr{L}_k$.
        Compute ρ_k and ϱ_k using (A.3).
        Compute Λ_k (A.4a) and $\Lambda_{0,k}$ (A.9a).
        Compute $\nabla_\vartheta \mathscr{C}_k$ and $\nabla_{x_0} \mathscr{C}_k$ using (A.4b) and (A.9b).
        $k \leftarrow k + 1$.
    end while
    Compute $\mathscr{C}_T$ using (A.2).
    Define $\nabla \mathscr{C}_T = [\nabla_\vartheta^\top \mathscr{C}_T, \nabla_{x_0}^\top \mathscr{C}_T]^\top$.
    Update $\hat{\vartheta}$, $\hat{x}_0$ using any first-order method.
end while
Return $\vartheta^\star = \hat{\vartheta}$ and $x_0^\star = \hat{x}_0$

A.3 Computational complexity

In the context of multi-step system identification, the computational complexity can easily explode as the multi-step horizon and parameter size increase [15]. Therefore, it is crucial to have a reliable algorithm whose complexity does not grow exponentially with these values. Thus, with the following theorem, we formally state the computational complexity of the proposed gradient computation algorithm for a given multi-step horizon and parameter size. As a measure of computational complexity, we consider the maximum number of operations needed to execute a given algorithm, expressed in Big O notation.

Theorem A.1 (Complexity analysis)

Let T be the length of the multi-step horizon considered in the system identification process, n_x the number of states, and $n_\vartheta = n_\theta + n_\omega$ the total number of parameters in the estimation model (2.3). The computational complexity required to compute the gradient by iterating the dynamical system (A.4) scales linearly with the length of the multi-step horizon T and the parameter size, exhibiting a computational complexity of $\mathcal{O}(Tn_x^2 n_\vartheta)$.

Proof. The proof follows from basic notions on matrix computation complexities [116]. Given two matrices, $A \in \mathbb{R}^{m,n}$ and $B \in \mathbb{R}^{n,o}$, the required computational complexity to perform the multiplication AB is $\mathcal{O}(mno)$, while the required computational complexity to perform the sum $A + A$ is $\mathcal{O}(mn)$. Similarly, given $C \in \mathbb{R}^{o,q}$, the required computational complexities required to perform $(AB)C$ and $A(BC)$ are $\mathcal{O}(mno + moq) = \mathcal{O}(mo(n + q))$ and $\mathcal{O}(noq + mnq) = \mathcal{O}(nq(o + m))$, respectively.

One iteration of (A.4) involves, in order, the multiplication $m_1 = \mathcal{J}_k^{x/x}\Lambda_{k-1}$, the sum $\Lambda_k = m_1 + \mathcal{J}_k^{x/\theta}$, the multiplication $m_2 = \Lambda_k^\top \rho_k$, and the sum $\nabla_\theta \mathscr{C}_k = \nabla_\theta \mathscr{C}_{k-1} + m_2 + \rho_k$ with complexities $\mathcal{O}(n_x^2 n_\theta)$, $\mathcal{O}(n_x n_\theta)$ $\mathcal{O}(n_x n_\theta)$, $\mathcal{O}(2n_\theta)$, respectively. Thus, the overall complexity for one iteration of (A.4) is $\mathcal{O}(n_x^2 n_\theta + 2n_x n_\theta + 2n_\theta) = \mathcal{O}(n_x^2 n_\theta)$. It follows that, for T iterations, the required complexity is $\mathcal{O}(Tn_x^2 n_\theta)$.

Note that, while n_x is typically fixed and depends on the system (2.1) under analysis, T and n_ϑ may vary, depending on the specific requirements of the problem. For instance, n_ϑ can easily increase, as it depends on the size of ω for the selected black-box model δ.

It is important to remark that, while (A.4) and (A.9) result from the application of forward AD in the context of multi-step system identification, a similar result can be obtained via the application of backward AD, as discussed next.

Remark A.2 (Backward AD in system identification)**.** *Forward AD provides a more intuitive and flexible representation of the gradient as it evolves forward in time with the model predictions, which enables the stability analysis in Section B.1. In contrast, backward AD requires processing the entire horizon before the gradient can be computed via reverse-time adjoint calculations [114]. Although this result is not included in this paper, we note for completeness that the computational complexity of the gradient computation via backward AD is $\mathcal{O}(T(n_x^2 + n_\vartheta n_x))$, and it can be derived using a reasoning similar to the one reported in the proof of Theorem A.1. Therefore, we remark that backward AD offers a valid alternative and a complementary perspective on the gradient dynamics, and is generally more efficient for high-dimensional problems with many parameters to identify, such as in the case of large black-box models δ.*

Concluding discussion

Appendix A introduced the computational backbone of the proposed identification framework, detailing how multi-step system identification problems can be efficiently solved through first-order optimization and automatic differentiation. By reformulating the recursive gradient propagation into LPV sensitivity equations, the appendix established a unified and scalable approach for computing exact gradients in nonlinear dynamical systems. This formulation bridges system identification and neural network optimization, revealing how tools traditionally used in learning theory can be adapted to physically grounded models.

From a theoretical standpoint, the results highlight that the overall computational complexity grows linearly with both the prediction horizon and the number of parameters, ensuring tractability for large-scale problems. From a methodological perspective, this analysis clarifies the interplay between multi-step propagation and gradient dynamics, setting the stage for the subsequent stability investigation in Appendix B. Together, these developments provide a rigorous and efficient foundation for implementing first-order methods in physics-informed system identification.

Appendix B

Stability analysis of multi-step gradients

This appendix formally addresses the "exploding gradient" issue: via a stability analysis of the LPV equations derived in appendix A, it proposes conditions for a reliable and efficient optimization and identification process for dynamical systems. In the final part of the appendix, we will show through simulation results that the proposed method is both effective and efficient, making it a promising tool for future research and applications in nonlinear system identification and non-convex optimization. The content of this Appendix is based on the published paper [110].

This appendix is organized as follows. Section B.1 addresses the problem of exploding gradients and introduces regularization strategies. Section B.2 presents a simple numerical example that illustrates the practical implications of complexity and gradient stability, including a population dynamics case study. Finally, Section B.3 draws some concluding remarks and summarizes the main insights of the contents of this appendix.

B.1 Non-exploding gradient

In the context of first-order methods, it is important to ensure the stability of the gradients used for the optimization. In this section, we investigate this issue by exploiting the dynamical formulation (A.4) to analyze the stability of the gradient, useful to ensure that gradients do not grow unboundedly and ensure a reliable identification. In this context, we offer a comprehensive approach grounded in systems theory to ensure an efficient stability characterization of the gradient dynamics. While the analysis is carried out for θ, we remark that the same reasoning with analogous results also applies to the gradient for the initial condition. First, the concept of non-exploding (or stable) gradient is formally defined as follows.

Definition B.1 (Non-exploding gradient)**.** *The multi-step gradient $\nabla_\theta \mathscr{C}_T$ is said to be non-exploding (or stable) if and only if there exists a finite constant μ_1 such that it satisfies*

$$\sup_{k\in[0,T]} \|\nabla_\theta \mathscr{C}_k\|_2 \leq \mu_1. \tag{B.1}$$

Definition B.1 follows from basic stability notions. For the gradient to be classified as non-exploding, it must have a finite "gain" throughout its evolution along the multi-step horizon T, described by (A.4). Thus, it is possible to establish a more specific condition for gradient stability by formalizing the link between multi-step gradient dynamics defined by the LPV system (A.4) and BIBO stability (see e.g., [117]), as stated with the following theorem.

Theorem B.1 (gradient stability)

The multi-step gradient $\nabla_\theta \mathscr{C}_T$ is non-exploding according to Definition B.1 if and only if the associated LPV system (A.4) is BIBO stable, i.e., there exists a finite constant μ_2 such that

$$\sum_{i=j}^{k-1} \|\Lambda_i^\top \rho_i + \varrho_i\|_2 \leq \mu_2, \tag{B.2}$$

for all $k \in [1, T]$, j with $k \geq j + 1$.

Proof. We observe that (A.4b) can be seen as a linear, time-varying state-space system

$$\begin{aligned} x_{k+1} &= A_k x_k + B_k u_k \\ y_k &= C_k x_k + D_k u_k \end{aligned} \tag{B.3}$$

with states $\nabla_\theta \mathscr{C}_k \in \mathbb{R}^{n_\theta}$, and

$$\begin{aligned} A_k = A \doteq I_{n_\theta} \in \mathbb{R}^{n_\theta, n_\theta},\ B_k \doteq (\Lambda_k^\top \rho_k + \varrho_k) \in \mathbb{R}^{n_\theta}, \\ C_k = C \doteq I_{n_\theta} \in \mathbb{R}^{n_\theta, n_\theta},\ D_k = D \doteq \mathbf{0}_{n_\theta} \in \mathbb{R}^{n_\theta}, \end{aligned} \tag{B.4}$$

with constant (bounded) input $u_k = 1 \in \mathbb{R}$, $\forall k \in [0, T]$. By analyzing the BIBO stability properties of the LTV system defined by (B.4), we can study the exploding gradient phenomenon and obtain conditions under which the multi-step gradient remains bounded, i.e., condition (B.1) is satisfied. The input-output behavior of (A.4b) is specified by the unit-pulse response

$$G(k, j) = C_k \Phi(k, j + 1) B_j,\ k \geq j + 1$$

with $\Phi(k, j)$ the transition matrix [117, Chapter 20], defined for $k > j$ as

$$\Phi(k, j) = \begin{cases} A_{k-1} A_{k-2} \dots A_j & k \geq j + 1 \\ I & k = j. \end{cases}$$

Stability results are characterized in terms of boundedness properties of $G(k, j)$. From [117, Theorem 27.2] we have that the linear state equation (A.4b) is uniformly BIBO stable if and only if there exists a finite constant μ_2 such that the unit-pulse response satisfies

$$\sum_{i=j}^{k-1} \|G(k, i)\| \leq \mu_2$$

for all k, j with $k \geq j + 1$. Notice that for our system defined by (B.4), we have $\Phi(k, j) = I_{n_\theta}$, $\forall k, j$, $G(k, j) = B_j$, $k \geq j + 1$, that yields (B.2), considering the multi-step horizon $[0, T]$. □

Consequently, it follows from Theorem B.1 and (A.4a) that certain conditions on the system's predicted trajectory can provide sufficient guarantees for the gradient to be bounded over the multi-step horizon T, as detailed in the following remark.

Remark B.1 (On the dynamical interpretation of Theorem B.1)**.** *The boundedness of the multi-step gradient recursion* (B.2) *can be intuitively related to the evolution of the predicted trajectory generated by the model dynamics* (2.3)*. Specifically, within the finite prediction horizon $k \in [0, T]$, condition* (B.2) *holds provided that the quantities Λ_i, ρ_i, and ϱ_i remain bounded for all $i \in [j, k-1]$ and all $k \in [1, T]$ with $k \geq j+1$. By Lipschitz continuity, the terms ρ_i and ϱ_i are guaranteed to be bounded, since the cost function $\mathscr{L}_k(\cdot)$, the prediction error e_k, and the output map $h(\cdot)$ are continuously differentiable. On the other hand, the term Λ_k may become unbounded, as its evolution follows the recursion* (A.4a)*. This recursion can be interpreted as a linear, time-varying state-space system with*

$$x_k = \mathrm{vec}(\Lambda_k) \in \mathbb{R}^{n_x n_\theta}, \quad u_k = \mathrm{vec}(\mathscr{J}_k^{x/\theta}) \in \mathbb{R}^{n_x n_\theta}, \quad A_k = I_{n_\theta} \otimes \mathscr{J}_k^{x/x} \in \mathbb{R}^{n_x n_\theta, n_x n_\theta},$$

where vec$(\cdot)$ denotes the vectorization of a matrix, obtained by stacking its columns into a single column vector. Its stability is therefore directly linked to the Jacobians $\mathscr{J}_k^{x/x}$, which represent the linearization of the nonlinear system under analysis around the predicted trajectory $\{\hat{x}_0, \dots, \hat{x}_T\}$, obtained with the current values of $\hat{\theta}$ and $\hat{x}_0$. This establishes a direct connection between condition (B.2) *and the stability of the predicted trajectories of the system (see, e.g., [117, Chapter 22]). Therefore, as the boundedness of the gradient recursion ultimately depends on the norm of the propagated Jacobians, a sufficient condition for the gradient to remain bounded is that the predicted trajectory evolves within a compact region of the state space where the system dynamics are smooth and the Jacobians are uniformly bounded over the considered multi-step horizon.*

Following this discussion, a practical implication is that, whenever some physical insight is available regarding the stability of system (C.1), it is possible to incorporate this information into the cost function (A.2) by introducing suitable penalty terms that promote stable gradient evolution during identification. Clearly, knowing the parameter values that lead to instability would be beneficial, as this enables directly imposing

constraints on the parameter values. However, it should be remarked that, in most cases, such values are not known to the user. In general, only the values of the states at which the system does not show instability are known, leading to state-dependent penalty terms, as detailed in the following remark.

Remark B.2 (Trajectory stability via state barriers)**.** *Assume that the underlying system is known to be stable within the interval* $x \in \mathcal{X} \doteq \left\{x_i^{lb} \leq \hat{x}_i \leq x_i^{ub},\ i = [1, n_x]\right\}$. *Then, the optimization can be steered in order to remain in this safe area by expressing in* (A.2) *a penalty term defined by the composition of two barrier functions, i.e.,*

$$p(\hat{x}_k, \theta) \doteq \lambda \| e^{\alpha(\hat{x}_k - x^{ub})} \|_2^2 + \lambda \| e^{\alpha(x^{lb} - \hat{x}_k)} \|_2^2, \tag{B.5}$$

with $\lambda, \alpha \in \mathbb{R}$ *tunable parameters.*

As a consequence, the predicted state variables are encouraged to stay within the specific intervals where the trajectories are known to be stable, as shown in the example proposed the following.

B.2 A population dynamics example

In this section, a numerical example is provided. In particular, the results of Theorem B.1 are tested through simulation for the identification of the parameters of a population dynamics model.

Let us consider a population dynamics model described by the discrete-time logistic map [118], [119], i.e.,

$$x_{k+1} = \theta x_k (1 - x_k),$$

where $x_k \in \mathbb{R}$ represents the ratio of the existing population to the maximum possible population at time step k, restricted to the interval [0,1], and $\theta \in \mathbb{R}$ is the parameter representing the growth rate. The behavior of the logistic map depends crucially on θ. In particular, while for $0 \leq \theta \leq 4$ we have that x_k converges, oscillates, or exhibits chaotic behavior in [0,1], for $\theta > 4$ it leaves the "safe" interval [0,1] and diverges, for almost all initial conditions.

In this example, we aim at identifying the growth rate θ that characterizes a given, stable, nominal population evolution over a horizon $T = 10^4$, i.e., $\{\tilde{x}_0, \dots, \tilde{x}_T\}$. Since the state variable represents the ratio between the current and maximum population, it is reasonable to assume that the values of the states where the system exhibits stable behavior are available. Thus, according to Remark B.2, it is possible to rely on a penalty term of the form (B.5) to bind the predicted trajectories in the desired interval, i.e., [0,1], thus ensuring stable gradients and allowing the identification of θ. In particular, we will exploit the following exponential barrier function having $\lambda = 10$ and $\alpha = 100$, i.e.,

$$p(\hat{x}_k) \doteq 10 e^{100(\hat{x}_k - 1)} + 10 e^{100(-\hat{x}_k)}. \tag{B.6}$$

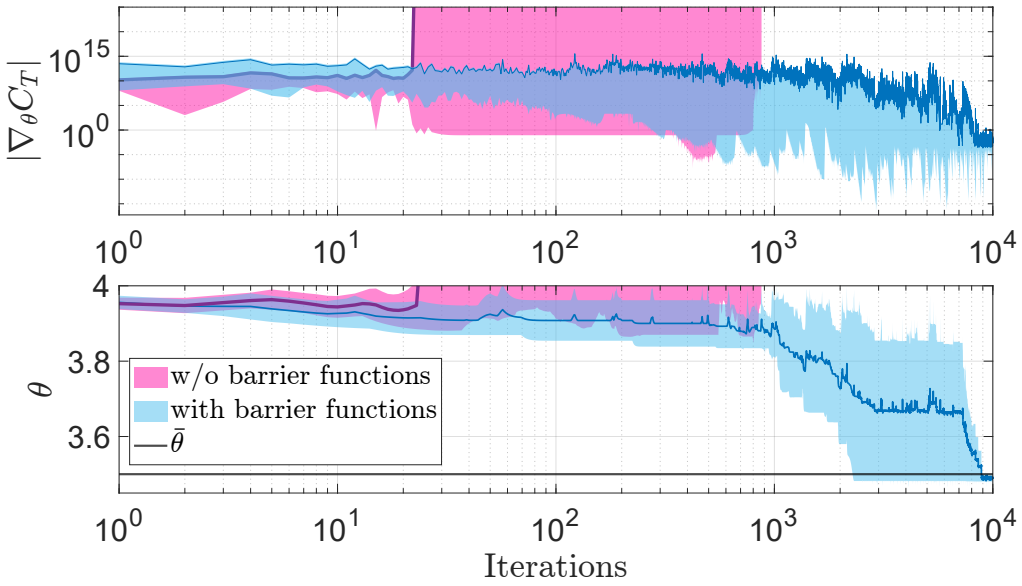


Figure B.1: Effect of barrier functions on mitigating exploding gradients for $N = 50$ different system initial conditions, represented with ± 1 standard deviation bands around the mean trajectories.

Let us consider a case with true parameter $\bar{\theta} = 3.5$, and $\hat{\theta} = 3.95 + \mathcal{N}(0,10^{-4})$ as the current estimation of the identification algorithm, aiming to identify the true value. This is a "critical" area for the parameters, very close to the values of instability ($\theta > 4$), and in the region where the system exhibits chaotic behavior. Therefore, it is likely for the gradient, when no barrier functions were used, to update the estimated parameter in the direction of a possible local minimum near the values of $\theta > 4$, causing gradient explosion. Fig. B.1 shows the evolution of the estimated parameters along with the computed gradient for $N = 50$ randomly selected system initial conditions $x_0 \sim \mathcal{U}(0,1)$, when no barrier functions are used, and when the barrier function proposed in (B.6) is adopted in the loss function (A.2). One can notice that, when no barrier functions are used, the gradient explodes when the estimated parameter reaches the finite-time instability zone, i.e., $\hat{\theta} > 4$. On the other hand, the use of a barrier function (B.6) helps the gradient to avoid updating the parameters to values that cause instability, until reaching a neighborhood of the true parameter value.

B.3 Concluding discussion

The results presented in this appendix complement the analysis of Appendix A by showing how the gradient dynamics derived from the proposed multi-step optimization framework provide further insight into the optimization process. While Appendix A established how the recursive formulation of the gradient can be exploited to improve

computational efficiency and scalability, the present analysis demonstrates how such efficiency can be combined with conditions that ensure the gradients remain bounded along the optimization horizon. In particular, the stability characterization developed here offers a rigorous systems-theoretic perspective that goes beyond heuristic techniques such as gradient clipping, and highlights how the boundedness of the predicted trajectories directly impacts the boundedness of the multi-step gradient.

From a practical viewpoint, these insights suggest that incorporating physical or structural prior knowledge into the optimization process is an effective strategy to mitigate instability. By enforcing stability-aware penalty terms or barrier functions, the identification can be steered toward trajectories where the gradient remains well conditioned, thereby improving both robustness and reliability of the overall procedure. Taken together, the contributions of Appendices A and B offer a unified view: the recursive gradient formulation ensures tractability at scale, while the gradient stability analysis guarantees that such tractability is not compromised by numerical or dynamical instabilities. This combination lays the foundation for reliable and scalable identification of nonlinear dynamical systems with first-order methods, and opens the way to further extensions where stability considerations are embedded directly into learning and control-oriented identification frameworks.

Appendix C

Kernel approximation theory

In this Appendix, we provide a brief introduction to nonlinear function approximation using kernels, which represents the core foundation for the main results presented in Chapter 5. First, we introduce two key definitions related to kernels.

Definition C.1 (Positive definite kernel [67])**.** *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 for all $n \in \mathbb{N}$, $x_1, \ldots, x_n \in \mathcal{X}$, $c_1, \ldots, c_n \in \mathbb{R}$.

Definition C.2 (Reproducing kernel Hilbert space [97], [120])**.** *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:*

a) For any $x \in \mathcal{X}$, $\kappa(\cdot, x) \in \mathcal{H}$.

b) The reproducing property holds, i.e., for any $x \in \mathcal{X}$, $h \in \mathcal{H}$, $\langle h(\cdot), \kappa(\cdot, x) \rangle_{\mathcal{H}} = h(x)$.

From the reproducing property in Definition C.2.b, 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

$$\kappa(x, x') = \langle \kappa(\cdot, x), \kappa(\cdot, x') \rangle_{\mathcal{H}}.$$

According to the Moore–Aronszajn theorem [120], we know that every positive definite kernel κ uniquely defines a RKHS $\mathcal{H}$ in which κ serves as the reproducing kernel. Conversely, each RKHS is associated with a unique positive definite kernel. Common choices for the kernel function $\kappa(x, x')$ include the Gaussian (radial basis function) kernel $\kappa(x, x') = \exp(-\|x - x'\|^2/2\sigma^2)$, the Laplacian kernel $\kappa(x, x') = \exp(-\|x - x'\|/\sigma)$,

the polynomial kernel $\kappa(x, x') = (x^\top x' + c)^d$, and the linear kernel $\kappa(x, x') = x^\top P x'$, where σ, c, d, and $P \succeq 0$ are hyperparameters.

Given a kernel κ, we next consider a nonlinear input-output relation given by an unknown nonlinear function $g : \mathscr{X} \to \mathbb{R}$

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

where $x \in \mathscr{X}$, $y \in \mathbb{R}$ are the input and output, respectively, g is assumed to belong to the native RKHS $\mathscr{H}$ 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 $\mathscr{D} = \{(x_1, y_1), \dots, (x_T, y_T)\}$ be a sequence of given T input-output data, collected from the system (C.1). The goal is to find an estimate $\hat{g}$ of 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 $\mathscr{D}$ involves minimizing a loss function that combines a quadratic data-fit term (i.e., the prediction error) with a regularization term. Hence, the unknown function g can be estimated by solving the following well-known kernel ridge regression (KRR) [99]

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

where $\gamma \in \mathbb{R}$ is a trade-off weight that balances data-fit and regularization, and $\|g\|_{\mathscr{H}} = \sqrt{\langle g, g \rangle_{\mathscr{H}}}$ is the norm in $\mathscr{H}$, introduced by the inner product $\langle \cdot, \cdot \rangle_{\mathscr{H}}$. Then, the *representer theorem* [67] guarantees the uniqueness of the solution to (C.2), expressed as a sum of T basis functions determined by the kernel, with their contributions weighted by coefficients obtained through the solution of a system of linear equations (see also, e.g., [97]). In particular, given a positive-definite, real-valued kernel $\kappa : \mathscr{X} \times \mathscr{X} \to \mathbb{R}$, 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 $Y \doteq [y_1, \dots, y_T]^\top$, the application of the representer theorem yields $\hat{g}$ in closed form as follows

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

Thus, considering (C.3) and solving the KRR (C.2) (see [99]), the weights vector $\omega = [\omega_1, \dots, \omega_T]$ is given by

$$\omega = (\mathbf{K} + \gamma \mathbf{I}_T)^{-1} Y,$$

with $\mathbf{I}_T$ denoting the $T \times T$ identity matrix.

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 fundamental result, the Chapter 5 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.

Bibliography

[1] L. Ljung, *System Identification: Theory for the User*, 2nd. Upper Saddle River: Prentice Hall PTR, 1999, ISBN: 0-13-656695-2.

[2] L. Ljung, «Perspectives on system identification,» *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 7172–7184, 2008.

[3] P. G. Hamel and R. V. Jategaonkar, «Evolution of flight vehicle system identification,» *Journal of aircraft*, vol. 33, no. 1, pp. 9–28, 1996.

[4] B. A. H. Vicente, S. S. James, and S. R. Anderson, «Linear system identification versus physical modeling of lateral–longitudinal vehicle dynamics,» *IEEE Transactions on Control Systems Technology*, vol. 29, no. 3, pp. 1380–1387, 2020.

[5] M. Baumann, C. Weissinger, and H.-G. Herzog, «System identification and modeling of an automotive bidirectional DC/DC converter,» in *2019 IEEE Vehicle Power and Propulsion Conference (VPPC)*, IEEE, 2019, pp. 1–5.

[6] D. Slaifstein, F. M. Ibanez, and K. Siwek, «Supercapacitor modeling: A system identification approach,» *IEEE Transactions on Energy Conversion*, vol. 38, no. 1, pp. 192–202, 2022.

[7] M. Daneker, Z. Zhang, G. E. Karniadakis, and L. Lu, «Systems biology: Identifiability analysis and parameter identification via systems-biology-informed neural networks,» in *Computational Modeling of Signaling Networks*, Springer, 2023, pp. 87–105.

[8] J. Hurrell, G. A. Meehl, D. Bader, T. L. Delworth, B. Kirtman, and B. Wielicki, «A unified modeling approach to climate system prediction,» *Bulletin of the American Meteorological Society*, vol. 90, no. 12, pp. 1819–1832, 2009.

[9] J. D. Farmer et al., «A complex systems approach to constructing better models for managing financial markets and the economy,» *The European Physical Journal Special Topics*, vol. 214, no. 1, pp. 295–324, 2012.

[10] M. Gevers, «Identification for control: From the early achievements to the revival of experiment design,» *European journal of control*, vol. 11, no. 4-5, pp. 335–352, 2005.

[11] W. Turner, A. Staino, and B. Basu, «Residential HVAC fault detection using a system identification approach,» *Energy and Buildings*, vol. 151, pp. 1–17, 2017.

[12] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, «Characterising the digital twin: A systematic literature review,» *CIRP journal of manufacturing science and technology*, vol. 29, pp. 36–52, 2020.

[13] L. Ljung, «Perspectives on system identification,» *Annual Reviews in Control*, vol. 34, no. 1, pp. 1–12, 2010.

[14] J. Schoukens, M. Vaes, and R. Pintelon, «Linear system identification in a nonlinear setting: Nonparametric analysis of the nonlinear distortions and their impact on the best linear approximation,» *IEEE Control Systems Magazine*, vol. 36, no. 3, pp. 38–69, 2016.

[15] J. Schoukens and L. Ljung, «Nonlinear system identification: A user-oriented road map,» *IEEE Control Systems Magazine*, vol. 39 (6), pp. 28–99, 2019.

[16] A. Svensson and T. B. Schön, «A flexible state–space model for learning nonlinear dynamical systems,» *Automatica*, vol. 80, pp. 189–199, 2017.

[17] P. Mattsson, D. Zachariah, and P. Stoica, «Identification of cascade water tanks using a PWARX model,» *Mechanical systems and signal processing*, vol. 106, pp. 40–48, 2018.

[18] A. Chiuso and G. Pillonetto, «System identification: A machine learning perspective,» *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 2, pp. 281–304, 2019.

[19] K. Schittkowski, *Numerical data fitting in dynamical systems: a practical introduction with applications and software*. Springer Science & Business Media, 2002, vol. 77.

[20] W. Greblicki and M. Pawlak, *Nonparametric system identification*. Cambridge University Press, 2008, vol. 1.

[21] M. Zorzi and A. Chiuso, «Sparse plus low rank network identification: A nonparametric approach,» *Automatica*, vol. 76, pp. 355–366, 2017.

[22] G. Pillonetto, F. Dinuzzo, T. Chen, G. De Nicolao, and L. Ljung, «Kernel methods in system identification, machine learning and function estimation: A survey,» *Automatica*, vol. 50 (3), pp. 657–682, 2014.

[23] M. Schoukens and F. G. Scheiwe, «Modeling nonlinear systems using a Volterra feedback model,» in *Workshop on nonlinear system identification benchmarks*, 2016.

[24] A. Dalla Libera, R. Carli, and G. Pillonetto, «Kernel-based methods for Volterra series identification,» *Automatica*, vol. 129, 2021.

[25] B. Mavkov, M. Forgione, and D. Piga, «Integrated neural networks for nonlinear continuous-time system identification,» *IEEE Control Systems Letters*, vol. 4 (4), pp. 851–856, 2020.

[26] F. Bonassi, M. Farina, J. Xie, and R. Scattolini, «On recurrent neural networks for learning-based control: Recent results and ideas for future developments,» *Journal of Process Control*, vol. 114, pp. 92–104, 2022.

[27] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, «Physics-informed machine learning,» *Nature Reviews Physics*, vol. 3, no. 6, pp. 422–440, 2021.

[28] A. M. Salih et al., «A perspective on explainable artificial intelligence methods: SHAP and LIME,» *Advanced Intelligent Systems*, vol. 7, no. 1, p. 2 400 304, 2025.

[29] G. Riva and S. Formentin, «Toward eXplainabile Data-Driven Control (XDDC): The property-preserving framework,» *IEEE Control Systems Letters*, vol. 8, pp. 478–483, 2024.

[30] M. Farina and L. Piroddi, «Simulation error minimization identification based on multi-stage prediction,» *International Journal of Adaptive Control and Signal Processing*, vol. 25 (5), pp. 389–406, 2011.

[31] E. Terzi, L. Fagiano, M. Farina, and R. Scattolini, «Learning multi-step prediction models for receding horizon control,» in *2018 European Control Conference (ECC)*, 2018, pp. 1335–1340.

[32] Q. Zhang, «Nonlinear system identification with output error model through stabilized simulation,» *IFAC Proceedings Volumes*, vol. 37, no. 13, pp. 501–506, 2004.

[33] L. Piroddi, «Simulation error minimisation methods for NARX model identification,» *International Journal of Modelling, Identification and Control*, vol. 3, no. 4, pp. 392–403, 2008.

[34] Y. Nesterov, *Lectures on convex optimization*. Springer, 2018, vol. 137.

[35] M. Ahookhosh, «Accelerated first-order methods for large-scale convex optimization: Nearly optimal complexity under strong convexity,» *Mathematical Methods of Operations Research*, vol. 89, no. 3, pp. 319–353, 2019.

[36] M. Teboulle, «A simplified view of first order methods for optimization,» *Mathematical Programming*, vol. 170, no. 1, pp. 67–96, 2018.

[37] H. Min, R. Vidal, and E. Mallada, «On the convergence of gradient flow on multi-layer linear models,» in *International Conference on Machine Learning*, PMLR, 2023, pp. 24 850–24 887.

[38] J. C. Aguero, G. C. Goodwin, and J. I. Yuz, «System identification using quantized data,» in *2007 46th IEEE Conference on Decision and Control*, IEEE, 2007, pp. 4263–4268.

[39] O. M. Sleem and C. M. Lagoa, «Parsimonious system identification from fragmented quantised measurements,» *International Journal of Control*, vol. 97, no. 8, pp. 1770–1779, 2024.

[40] B. Yılmaz, K. Bekiroglu, C. Lagoa, and M. Sznaier, «A randomized algorithm for parsimonious model identification,» *IEEE Transactions on Automatic Control*, vol. 63, no. 2, pp. 532–539, 2018.

[41] H. S. Huntley and G. J. Hakim, «Assimilation of time-averaged observations in a quasi-geostrophic atmospheric jet model,» *Climate dynamics*, vol. 35, pp. 995–1009, 2010.

[42] A. Amin and M. Mourshed, «Weather and climate data for energy applications,» *Renewable and Sustainable Energy Reviews*, vol. 192, p. 114 247, 2024.

[43] S. M. Kidwell and A. Tomasovych, «Implications of time-averaged death assemblages for ecology and conservation biology,» *Annual Review of Ecology, Evolution, and Systematics*, vol. 44, no. 1, pp. 539–563, 2013.

[44] P. J. Wangersky, «Lotka-Volterra population models,» *Annual Review of Ecology and Systematics*, vol. 9, pp. 189–218, 1978.

[45] S. Nakagawa and R. P. Freckleton, «Model averaging, missing data and multiple imputation: A case study for behavioural ecology,» *Behavioral Ecology and Sociobiology*, vol. 65, pp. 103–116, 2011.

[46] D. Givoly and D. Palmon, «Timeliness of annual earnings announcements: Some empirical evidence,» *Accounting review*, pp. 486–508, 1982.

[47] C. Grossmann, C. N. Jones, and M. Morari, «System identification via nuclear norm regularization for simulated moving bed processes from incomplete data sets,» in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, IEEE, 2009, pp. 4692–4697.

[48] H. Raghavan, A. K. Tangirala, R. B. Gopaluni, and S. L. Shah, «Identification of chemical processes with irregular output sampling,» *Control Engineering Practice*, vol. 14, no. 5, pp. 467–480, 2006.

[49] A. J. Isaksson, «Identification of ARX-models subject to missing data,» *IEEE Transactions on Automatic Control*, vol. 38, no. 5, pp. 813–819, 1993.

[50] Z. Liu, A. Hansson, and L. Vandenberghe, «Nuclear norm system identification with missing inputs and outputs,» *Systems & Control Letters*, vol. 62, no. 8, pp. 605–612, 2013.

[51] S. K. Varanasi and P. Jampana, «Nuclear norm subspace identification of continuous time state–space models with missing outputs,» *Control Engineering Practice*, vol. 95, p. 104 239, 2020.

[52] R. B. Gopaluni, T. B. Schön, and A. G. Wills, «Particle filter approach to nonlinear system identification under missing observations with a real application,» *IFAC Proceedings Volumes*, vol. 42, no. 10, pp. 810–815, 2009.

[53] R. B. Gopaluni, «Nonlinear system identification under missing observations: The case of unknown model structure,» *Journal of Process Control*, vol. 20, no. 3, pp. 314–324, 2010.

[54] T. Demeester, «System identification with time-aware neural sequence models,» in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 3757–3764.

[55] Z. Yuan, X. Ban, Z. Zhang, X. Li, and H.-N. Dai, «ODE-RSSM: Learning stochastic recurrent state space model from irregularly sampled data,» in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, 2023, pp. 11 060–11 068.

[56] G. Pillonetto, A. Aravkin, D. Gedon, L. Ljung, A. H. Ribeiro, and T. B. Schön, «Deep networks for system identification: A survey,» *Automatica*, vol. 171, 2025.

[57] J. Fan, T. W. Chow, and S. J. Qin, «Kernel-based statistical process monitoring and fault detection in the presence of missing data,» *IEEE Transactions on Industrial Informatics*, vol. 18, no. 7, pp. 4477–4487, 2021.

[58] Y. Yu, V. O. Li, J. C. Lam, and K. Chan, «GCN-ST-MDIR: Graph convolutional network-based spatial-temporal missing air pollution data pattern identification and recovery,» *IEEE Transactions on Big Data*, vol. 9, no. 5, pp. 1347–1364, 2023.

[59] W. Quaghebeur, I. Nopens, and B. De Baets, «Incorporating unmodeled dynamics into first-principles models through machine learning,» *IEEE Access*, vol. 9, pp. 22 014–22 022, 2021.

[60] M. Mammarella, C. Donati, F. Dabbene, C. Novara, and C. Lagoa, «A blended physics-based and black-box identification approach for spacecraft inertia estimation,» in *2024 IEEE 63rd Conference on Decision and Control (CDC)*, 2024, pp. 8282–8287.

[61] M. Zakwan, L. Di Natale, B. Svetozarevic, P. Heer, C. N. Jones, and G. F. Trecate, «Physically consistent neural ODEs for learning multi-physics systems,» *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 5855–5860, 2023.

[62] Y. Liu, R. Tóth, and M. Schoukens, «Physics-guided state-space model augmentation using weighted regularized neural networks,» *IFAC-PapersOnLine*, vol. 58, no. 15, pp. 295–300, 2024, 20th IFAC Symp. on System Identification, ISSN: 2405-8963.

[63] K. Kaheman, E. Kaiser, B. Strom, J. N. Kutz, and S. L. Brunton, «Learning discrepancy models from experimental data,» *58th IEEE Conf. on Decision and Control*, 2019.

[64] J. Brynjarsdóttir and A. O'Hagan, «Learning about physical parameters: The importance of model discrepancy,» *Inverse problems*, vol. 30, no. 11, p. 114 007, 2014.

[65] M. Forgione, A. Muni, D. Piga, and M. Gallieri, «On the adaptation of recurrent neural networks for system identification,» *Automatica*, vol. 155, p. 111 092, 2023.

[66] 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 Systems Magazine*, vol. 43 (5), pp. 69–110, 2023.

[67] B. Schölkopf, R. Herbrich, and A. J. Smola, «A generalized representer theorem,» in *International conference on computational learning theory*, Springer, 2001, pp. 416–426.

[68] 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.

[69] 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, p. 112 409, 2025, ISSN: 0005-1098.

[70] C. Donati, M. Mammarella, F. Dabbene, C. Novara, and C. Lagoa, «Recovering nonlinear dynamics from non-uniform observations: A physics-based identification approach with practical case studies,» *Control Engineering Practice*, vol. 164, p. 106 411, 2025, ISSN: 0967-0661.

[71] C. Donati, M. Mammarella, G. Calafiore, F. Dabbene, C. M. Lagoa, and C. Novara, «A kernel-based approach to physics-informed nonlinear system identification,» *(see arxiv.org/abs/2509.07634) under review for: Transactions on Automatic Control*, 2025.

[72] J. Sjöberg et al., «Nonlinear black-box modeling in system identification: A unified overview,» *Automatica*, vol. 31, no. 12, pp. 1691–1724, 1995.

[73] M. Schmidt, G. Fung, and R. Rosales, «Fast optimization methods for l1 regularization: A comparative study and two new approaches,» in *18th European Conference on Machine Learning*, Springer, 2007, pp. 286–297.

[74] Z. R. Manchester and M. A. Peck, «Recursive inertia estimation with semidefinite programming,» in *AIAA Guidance, Navigation, and Control Conference*, 2017, p. 1902.

[75] A. C. B. de Oliveira, M. Siami, and E. D. Sontag, «Dynamics and perturbations of overparameterized linear neural networks,» in *2023 62nd IEEE Conference on Decision and Control (CDC)*, 2023.

[76] Z. Xu, H. Min, S. Tarmoun, E. Mallada, and R. Vidal, «Linear convergence of gradient descent for finite width over-parametrized linear networks with general initialization,» in *International Conference on Artificial Intelligence and Statistics*, 2023.

[77] L. Chizat and F. Bach, «On the global convergence of gradient descent for over-parameterized models using optimal transport,» *Advances in neural information processing systems*, vol. 31, 2018.

[78] I. A. Lomaka, «A possible approach to the identification of inertial parameters of large-sized space debris using a specialized nanosatellite,» in *Journal of Physics: Conference Series*, 2020.

[79] M. Schoukens, P. Mattson, T. Wigren, and J.-P. Noël, «Cascaded tanks benchmark combining soft and hard nonlinearities,» in *Workshop on Nonlinear System Identification Benchmarks*, 2016.

[80] R. Relan, K. Tiels, A. Marconato, and J. Schoukens, «An unstructured flexible nonlinear model for the cascaded water-tanks benchmark,» *IFAC-PapersOnLine*, vol. 50 (1), pp. 452–457, 2017.

[81] M. Brunot, A. Janot, and F. Carrillo, «Continuous-time nonlinear systems identification with output error method based on derivative-free optimisation,» *IFAC-PapersOnLine*, vol. 50 (1), pp. 464–469, 2017.

[82] R. Bellman and K. J. Åström, «On structural identifiability,» *Mathematical biosciences*, vol. 7, no. 3-4, pp. 329–339, 1970.

[83] E. T. Maddalena, P. Scharnhorst, and C. N. Jones, «Deterministic error bounds for kernel-based learning techniques under bounded noise,» *Automatica*, vol. 134, p. 109 896, 2021.

[84] S. G. Krantz and H. R. Parks, *The implicit function theorem: History, theory, and applications*. Springer Science & Business Media, 2002.

[85] R. Gribonval, R. Figueras i Ventura, and P. Vandergheynst, «A simple test to check the optimality of a sparse signal approximation,» *Signal processing*, vol. 86, no. 3, pp. 496–510, 2006.

[86] C. Novara, «Sparse identification of nonlinear functions and parametric set membership optimality analysis,» *IEEE Trans. on Automatic Control*, vol. 57, pp. 3236–3241, 2012.

[87] C. Novara, «Sparse identification of nonlinear functions and parametric set membership optimality analysis,» in *Proceedings of the 2011 American Control Conference*, 2011, pp. 663–668.

[88] J. D. Morningred, B. E. Paden, D. E. Seborg, and D. A. Mellichamp, «An adaptive nonlinear predictive controller,» *Chemical Engineering Science*, vol. 47, no. 4, pp. 755–762, 1992.

[89] B. De Moor, P. De Gersem, B. De Schutter, W. Favoreel, et al., «Daisy: A database for identification of systems,» *JOURNAL A*, vol. 38, no. 4, pp. 4–5, 1997.

[90] Y. Lu and B. Huang, «Robust multiple-model LPV approach to nonlinear process identification using mixture t distributions,» *Journal of Process Control*, vol. 24, no. 9, pp. 1472–1488, 2014.

[91] R. Gopaluni, «A particle filter approach to identification of nonlinear processes under missing observations,» *The Canadian Journal of Chemical Engineering*, vol. 86, no. 6, pp. 1081–1092, 2008.

[92] J. Deng and B. Huang, «Identification of nonlinear parameter varying systems with missing output data,» *AIChE Journal*, vol. 58, no. 11, pp. 3454–3467, 2012.

[93] X. Yang, X. Liu, and S. Yin, «Robust identification of nonlinear systems with missing observations: The case of state-space model structure,» *IEEE Transactions on Industrial Informatics*, vol. 15, no. 5, pp. 2763–2774, 2018.

[94] Å. Björck, *Numerical Methods for Least Squares Problems*. Society for Industrial and Applied Mathematics, 1996.

[95] O. Malcai, O. Biham, P. Richmond, and S. Solomon, «Theoretical analysis and simulations of the generalized Lotka-Volterra model,» *Physical Review E*, vol. 66, no. 3, p. 031 102, 2002.

[96] D. Ruppert, *The elements of statistical learning: data mining, inference, and prediction*. Taylor & Francis, 2004.

[97] G. Wahba, *Spline models for observational data*. SIAM, 1990.

[98] A. J. Smola and B. Schölkopf, «A tutorial on support vector regression,» *Statistics and computing*, vol. 14, pp. 199–222, 2004.

[99] C. Saunders, A. Gammerman, and V. Vovk, «Ridge regression learning algorithm in dual variables,» in *Proceedings of the Fifteenth International Conference on Machine Learning*, ser. ICML '98, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998, pp. 515–521, ISBN: 1558605568.

[100] T. B. Schön, A. Wills, and B. Ninness, «System identification of nonlinear state-space models,» *Automatica*, vol. 47 (1), pp. 39–49, 2011.

[101] R. Frigola, F. Lindsten, T. B. Schön, and C. E. Rasmussen, «Bayesian inference and learning in Gaussian process state-space models with particle MCMC,» *Advances in neural information processing systems*, vol. 26, 2013.

[102] E. A. Wan and R. Van Der Merwe, «The unscented Kalman filter for nonlinear estimation,» in *Proceedings of the IEEE 2000 adaptive systems for signal processing, communications, and control symposium*, 2000.

[103] S. Särkkä, «Unscented Rauch–Tung–Striebel smoother,» *IEEE transactions on Automatic Control*, vol. 53 (3), pp. 845–849, 2008.

[104] J. Akhtar, I. Ghous, M. Jawad, Z. Duan, I. U. Khosa, and S. Ahmed, «A computationally efficient unscented Kalman smoother for ameliorated tracking of subatomic particles in high energy physics experiments,» *Computer Physics Communications*, vol. 283, 2023.

[105] S. J. Julier and J. K. Uhlmann, «Unscented filtering and nonlinear estimation,» *Proceedings of the IEEE*, vol. 92 (3), pp. 401–422, 2004.

[106] H. M. Menegaz, J. Y. Ishihara, G. A. Borges, and A. N. Vargas, «A systematization of the unscented Kalman filter theory,» *IEEE Transactions on Automatic Control*, vol. 60 (10), pp. 2583–2598, 2015.

[107] H. J. Van Waarde and R. Sepulchre, «Kernel-based models for system analysis,» *IEEE Transactions on Automatic Control*, vol. 68, no. 9, pp. 5317–5332, 2022.

[108] S. Formentin, D. Piga, R. Tóth, and S. M. Savaresi, «Direct learning of LPV controllers from data,» *Automatica*, vol. 65, pp. 98–110, 2016.

[109] T. B. Hamdan, P. Coirault, G. Mercère, and T. Dairay, «Data enabled predictive control of LPV systems,» *Control Engineering Practice*, vol. 149, p. 105 969, 2024.

[110] C. Donati, M. Mammarella, F. Dabbene, C. Novara, and C. Lagoa, «A scalable, gradient-stable approach to multi-step, nonlinear system identification using first-order methods,» *IFAC-PapersOnLine*, vol. 59, no. 15, pp. 37–42, 2025, 6th IFAC Workshop on Linear Parameter Varying Systems LPVS 2025.

[111] T. P. Lillicrap and A. Santoro, «Backpropagation through time and the brain,» *Current Opinion in Neurobiology*, vol. 55, pp. 82–89, 2019, ISSN: 0959-4388.

[112] C. Donati, M. Mammarella, F. Dabbene, C. Novara, and C. Lagoa, «One-shot backpropagation for multi-step prediction in physics-based system identification,» *20th IFAC Symposium on System Identification (SYSID)*, 2024.

[113] L. Behera, S. Kumar, and A. Patnaik, «On adaptive learning rate that guarantees convergence in feedforward networks,» *IEEE Trans. on Neural Networks*, vol. 17, pp. 1116–1125, 2006.

[114] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind, «Automatic differentiation in machine learning: A survey,» *Journal of Machine Learning Research*, vol. 18, no. 153, pp. 1–43, 2018.

[115] L. Di Natale, M. Zakwan, B. Svetozarevic, P. Heer, G. Ferrari-Trecate, and C. N. Jones, «Stable linear subspace identification: A machine learning approach,» in *2024 European Control Conference (ECC)*, IEEE, 2024, pp. 3539–3544.

[116] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU press, 2013.

[117] W. J. Rugh, *Linear system theory*. Prentice-Hall, Inc., 1996.

[118] R. M. May, «Simple mathematical models with very complicated dynamics,» *Nature*, vol. 261, no. 5560, pp. 459–467, 1976.

[119] G.-C. Wu and D. Baleanu, «Discrete fractional logistic map and its chaos,» *Nonlinear Dynamics*, vol. 75, pp. 283–287, 2014.

[120] N. Aronszajn, «Theory of reproducing kernels,» *Transactions of the American mathematical society*, vol. 68 (3), pp. 337–404, 1950.

This Ph.D. thesis has been typeset by means of the TEX-system facilities. The typesetting engine was LuaLATEX. The document class was `toptesi`, by Claudio Beccari, with option `tipotesi=scudo`. This class is available in every up-to-date and complete TEX-system installation.