



MOX-Report No. 30/2025

**Online learning in bifurcating dynamic systems via SINDy and
Kalman filtering**

Rosafalco, L.; Conti, P.; Manzoni, A.; Mariani, S.; Frangi, A.

MOX, Dipartimento di Matematica
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

mox-dmat@polimi.it

<https://mox.polimi.it>

Online learning in bifurcating dynamic systems via SINDy and Kalman filtering

Luca Rosafalco^a, Paolo Conti^{a,b}, Andrea Manzoni^b, Stefano Mariani^a, Attilio Frangi^a

^a*Department of Civil and Environmental Engineering, Politecnico di Milano
Piazza L. da Vinci 32, 20133 - Milano (Italy)*

^b*MOX, Department of Mathematics, Politecnico di Milano
Piazza L. da Vinci 32, 20133 - Milano (Italy)*

Abstract

We propose the use of the Extended Kalman Filter (EKF) for online data assimilation and update of a dynamic model, preliminary identified through the Sparse Identification of Nonlinear Dynamics (SINDy). This data-driven technique may avoid biases due to incorrect modelling assumptions and exploits SINDy to approximate the system dynamics leveraging a predefined library of functions, where active terms are selected and weighted by a sparse set of coefficients. This results in a physically-sound and interpretable dynamic model allowing to reduce epistemic uncertainty often affecting machine learning approaches. Treating the SINDy model coefficients as random variables, we propose to update them while acquiring (possibly noisy) system measurements, thus enabling the online identification of time-varying systems. These changes can stem from, e.g., varying operational conditions or unforeseen events. The EKF performs model adaptation through joint state-parameters estimation, with the Jacobian matrices required to compute the model sensitivity inexpensively evaluated from the SINDy model formulation. The effectiveness of this approach is demonstrated through three case studies: (i) a Lotka-Volterra model in which all parameters simultaneously evolve during the observation period; (ii) a Selkov model where the system undergoes a bifurcation not seen during the SINDy training; (iii) a MEMS arch exhibiting a 1:2 internal resonance. The ability of EKF of recovering inactivated functional terms from the SINDy library, or discarding unnecessary contribution, is also highlighted. Based on the presented applications, this method shows strong promise for handling time-varying nonlinear dynamic systems possibly experiencing bifurcating behaviours.

1. Introduction

The increasing availability of data has driven significant advancements across various scientific and engineering domains, including structural health monitoring in civil engineering [1], aerodynamic control [2], and modeling the structural mechanics of Micro Electro-Mechanical Systems (MEMS) [3]. In all these fields, data-driven approaches have emerged as new paradigms to analyse, forecast, and control complex physical systems, setting the foundation of predictive Digital Twins (DTs) [4], nowadays enabling, e.g., predictive maintenance for civil infrastructures [5] or unmanned aerial vehicles [6].

The construction of predictive DTs, unlocking the possibility to operate online on a dynamical system, requires to assimilate incoming information on-the-fly. This is possible thanks to filtering techniques like the Kalman Filter (KF) [7], that provides the optimal way to assimilate information in a linear setting [8]. Nonlinear extensions of the filter, such as the Extended Kalman Filter (EKF) or the unscented Kalman filter can be used for nonlinear identification problems [9, 10]. A broader application of filtering techniques for DTs is hampered by challenges in filter tuning and errors in formulating the dynamic model employed in the predictor stage of the filter [11, 12]. While advanced filter tuning approaches have been recently proposed in the literature [13, 14, 15], here we focus on data-driven non-intrusive surrogates to face the improper modelling issue suffered by purely physics-based models [16, 17].

Email addresses: luca.rosafalco@polimi.it (Luca Rosafalco), paolo.conti@polimi.it (Paolo Conti), andrea1.manzoni@polimi.it (Andrea Manzoni), stefano.mariani@polimi.it (Stefano Mariani), attilio.frangi@polimi.it (Attilio Frangi)

Recent studies completely replaced physics-based models with Machine Learning (ML)-based models. For example, generative adversarial networks have been demonstrated successful in structural health monitoring [18] and damage detection for nonlinear dynamic system [19] in simulated environments. In a similar direction, the combination of ML-based models with KFs was recently explored. For instance, in [20], long short-term memory networks were used in the predictor phase of the filter, resulting in the automatic estimation of the process and measurement noise. The work of [21] demonstrated impressive estimation capabilities in nonlinear structural dynamics by employing dynamical variational autoencoders whose inference task was performed by the EKF. On the other hand, training Neural Networks (NNs) require large datasets and struggle with epistemic uncertainties [22], which refer to the inability of modelling a system response in scenarios where data are unavailable [23]. This is the case of civil structure responses in potential damage conditions [24], where only baseline (undamaged) data are available. Such uncertainties, combined with the lack of interpretability and of physical consistency, cause significant limitations to the practical application of ML models in real engineering contexts, potentially hindering their use in the experimental setting [25]. Moreover, pre-trained NNs generally cannot be updated online, potentially leading to considerable prediction errors in nonlinear dynamics. This is particularly problematic when highly nonlinear processes are involved, as dynamic contributions that were negligible during the initial NN training may emerge. For example, variations of input parameters can result in the development of vorticity in fluids [26] or resonance in mechanical structures [27], possibly leading to bifurcation phenomena that completely change the system dynamics. In a DT perspective, detecting at which configurations these regime transitions occur is essential, as these behaviours can significantly affect the operation of the system. We highlight physical consistency as a required condition to make ML models more robust.

In this work, we present a data-driven method for online learning and adaptation of a dynamical model, using assimilated data from the physical system. Specifically, we adopt the Sparse Identification of Nonlinear Dynamics (SINDy) technique [28] to construct the predictive model required in the filtering prediction stage; data assimilation is then carried out using an EKF. SINDy approximates the system dynamics from available data by constructing a nonlinear dynamic model that is parametrised by a sparse set of coefficients, determining the active terms from a predefined library of functions. The construction of robust, physically consistent, and interpretable models is thus enabled by SINDy thanks to its capacity to adopt a parsimonious description of system dynamics, consistent with the assumption that system models are governed by few important terms if an appropriate reference system is adopted, even when discontinuous nonlinearities are considered [29].

Differently from our previous work [30], here we do not only update system variables to track the evolution of states and parameters; rather, we evolve the predictive model itself by dealing with SINDy coefficients as learnable random variables. The outcome is an explainable data-driven model that can evolve online adapting to different scenarios. This allows to enrich a dynamic description of the physical phenomenon at hand as new data are assimilated, and to correct or even dropping terms that are incorrect or irrelevant. The capacity to assimilate data and modify online the SINDy model sets apart our work from previous applications of SINDy, for which training is typically done once in a preliminary offline phase.

With respect to [30], we no longer require a large amount of trajectories of the system for different parameter configurations to construct, a priori, a predictive model which accounts for parametric dependency. Recovering this type of data is not always possible or it may result extremely expensive, although possible [31]. Instead we can learn and incorporate this parametric dependency online, thus significantly enlarging the applicability of the method.

Thanks to the convenient non-intrusive system parametrisation provided by SINDy, the obtained framework can model parametric bifurcations, which is a fundamental and desirable goal. For example, the possibility of generalising the system dynamics over bifurcation parameter values was recently commented in [32], by adapting a NN-based meta-model of the system dynamics. However, this approach required to preliminarily collect trajectories under varying the properties of interest.

Other works combined the ideas underlying the SINDy technique with filtering. For instance, in [33, 34], a sparsity promoting UKF algorithm was proposed for nonlinear dynamic identification, employing pseudo-observations as described in [35], with the advantage of eliminating the need of any preliminary offline procedure; however, omitting SINDy for parameter initialisation makes the identification task considerably more challenging. In [36], the combination with a KF was proposed to mitigate the impact of noise on the identification of the SINDy model.

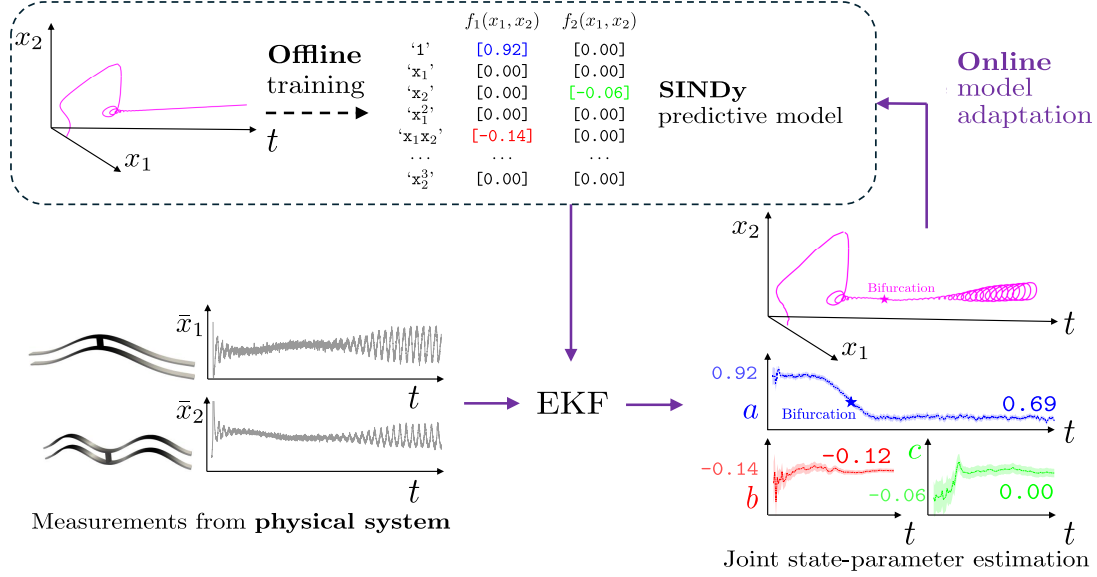


Figure 1: Online model adaptation combining SINDy and EKF. In the offline stage, a SINDy model is trained. In the online stage, this model is used in the predictor phase of the EKF to advance the system state over time. The EKF corrector phase then updates both the states and parameters of the model by assimilating measurements acquired on the physical system. The procedure can potentially recover inactivated functional terms from the SINDy library or, conversely, suppress unnecessary contributions.

Similarly to what we propose, in [37], SINDy parameters were evolved, but the use of a linear version of the KF excluded the possibility of joint estimation of system dynamics and model parameters, thus precluding nonlinear dynamics applications for which being able to closely track the evolution of the system is essential. By adopting the EKF, we have been instead able to dramatically widen the range of applications of the procedure targeting a collection of interesting and challenging problems. In particular, we treat a case in which all the parameters of a dynamic model vary simultaneously during the observation window; a further case in which the variation of one parameter causes the system to pass through an Hopf bifurcation; a final situation related with a real MEMS device featuring a 1:2 internal resonance and Neimark-Sacker bifurcations [38].

The remainder of the paper is organised as follows. In Sec. 2 the proposed procedure is discussed, focusing on EKF and SINDy equations, and highlighting how to combine them. A schematic representation of the proposed methodology, distinguishing between the offline SINDy training and its online update, is reported in Fig. 1. Case studies are discussed in Sec. 3, by addressing a time-varying Lokta-Volterra model (3.1), a Selkov model passing an Hopf bifurcations (Sec. 3.2) and a real MEMS device (Sec. 3.3). Further comments and future developments are reported in Sec. 4.

2. Methodology

We rely on a state-space representation of the dynamical system of interest. Such representation allows for the time evolution of the state variables describing the system dynamics, collected in the vector $\mathbf{x}(t) \in \mathbb{R}^n$; data assimilation exploits instead incoming observations of the system $\mathbf{y}(t_j) \in \mathbb{R}^o$, where o are the number of observations at t_j , and t_j with $j = 1, \dots, T$ are the discrete observation times. Observations and state components are treated as random variables, to account for modelling and measurement uncertainties. The following system of equations terms capture system evolution and observables:

$$\dot{\mathbf{x}}(t) = \frac{d}{dt}\mathbf{x}(t) = \mathbf{f}(\mathbf{x}(t), t) + \mathbf{q}_x, \quad t \in [t_0, t_T], \quad (1a)$$

$$\mathbf{y}(t_j) = \mathbf{h}(\mathbf{x}(t_j)) + \mathbf{r}, \quad t_j \in \{t_0, t_1, \dots, t_T\}, \quad (1b)$$

where: t is the time coordinate; $\mathbf{f} : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$ is the function describing the system time-varying dynamics; \mathbf{q}_x is the process noise vector; $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^o$ is a possibly nonlinear function mapping the state variables into the incoming measurements $\mathbf{y}(t_j)$. Both \mathbf{q}_x and \mathbf{r} are assumed to be distributed as white, zero mean stochastic processes, that is $\mathbf{q}_x \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_x)$ with diagonal covariance matrix $\mathbf{Q}_x \in \mathbb{R}^{n \times n}$, and $\mathbf{r} \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ with diagonal covariance matrix $\mathbf{R} \in \mathbb{R}^{o \times o}$, respectively. Eqs. (1a) and (1b) can be easily modified to include the dependence on a forcing term, as shown in [30]. It is worth noting that the continuous time dependence in Eq. (1a) enables to directly substitute \mathbf{f} with SINDy, while the discrete time dependence of the observables is consistent with digital recording acquisition.

A prediction-correction scheme is adopted: the quantities evolved through Eq. (1a), denoted with the superscript “−”, are the output of the prediction phase; these outputs will be then updated in the correction phase. The resulting quantities will be denoted with the superscript “+”. According to the extended version of the filter (EKF), the mean $\hat{\mathbf{x}} = \mathbb{E}[\mathbf{x}]$ and covariance $\mathbf{P} = \mathbb{E}[(\mathbf{x} - \hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}})^\top]$ of the state variables \mathbf{x} are propagated in time.

During the prediction stage, the values of $\hat{\mathbf{x}}^+(t_{j-1})$ and $\mathbf{P}^+(t_{j-1})$ at t_{j-1} are evolved to obtain $\hat{\mathbf{x}}^-(t_j)$ and $\mathbf{P}^-(t_j)$ at t_j , respectively. To integrate the continuous relation in Eq. (1a), different methods can be utilised. Due to its high accuracy, we have employed the fourth-order Runge Kutta integrator. More details on time integration can be found in Algorithm 1. To compute $\mathbf{P}^+(t_j)$, the Jacobian $\mathbf{F}^+(t_{j-1})$ of \mathbf{f} has to be computed according to:

$$\mathbf{F}^+(t_{j-1}) = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}^\top} \right|_{\hat{\mathbf{x}}^+(t_{j-1})}. \quad (2)$$

After getting $\hat{\mathbf{x}}^+(t_{j-1})$ and $\mathbf{P}^+(t_{j-1})$ in the prediction stage, their estimates are updated by assimilating the information from incoming measurements $\mathbf{y}(t_j)$ to get of $\hat{\mathbf{x}}^+(t_j)$ and $\mathbf{P}^+(t_j)$ at t_j . This is achieved by way of:

$$\mathbf{G}(t_j) = \mathbf{P}^-(t_j) \mathbf{H}^-(t_j)^\top (\mathbf{H}^-(t_j) \mathbf{P}^-(t_j) \mathbf{H}^-(t_j)^\top + \mathbf{R})^{-1}, \quad (3a)$$

$$\hat{\mathbf{x}}^+(t_j) = \hat{\mathbf{x}}^-(t_j) + \mathbf{G}(t_j)(\mathbf{y}(t_j) - \mathbf{h}(\hat{\mathbf{x}}^-(t_j), t_j)), \quad (3b)$$

$$\mathbf{P}^+(t_j) = (\mathbf{I} - \mathbf{G}(t_j) \mathbf{H}^-(t_j)) \mathbf{P}^-(t_j) (\mathbf{I} - \mathbf{G}(t_j) \mathbf{H}^-(t_j))^\top + \mathbf{G}(t_j) \mathbf{R} \mathbf{G}(t_j)^\top, \quad (3c)$$

where: $\mathbf{G}(t_j)$ is the Kalman gain at time t_j ; \mathbf{H} is the Jacobian of \mathbf{h} , computed as:

$$\mathbf{H}^-(t_j) = \left. \frac{\partial \mathbf{h}}{\partial \mathbf{x}^\top} \right|_{\hat{\mathbf{x}}^-(t_j)}. \quad (4)$$

In this work, we assume \mathbf{h} to be an operator selecting which state variable must be compared with the incoming observations. Alternatively, the same identification strategy here adopted for \mathbf{f} can be employed to identify not trivial \mathbf{h} , as already proposed in [30].

To avoid modelling issues [12], we identify \mathbf{f} directly from data. In particular, we consider the SINDy setup [28], where the function \mathbf{f} is approximated by a linear combination of p candidate functions included in a library $\Theta(\mathbf{x}(t)) = [\theta_1(\mathbf{x}(t)), \dots, \theta_p(\mathbf{x}(t))] \in \mathbb{R}^p$ that describes the system dynamics:

$$\mathbf{f}(\mathbf{x}(t), t) \approx \Xi^\top(t) \Theta^\top(\mathbf{x}(t)), \quad (5)$$

where $\Xi(t) = [\xi_1(t), \dots, \xi_n(t)] \in \mathbb{R}^{p \times n}$ is the matrix collecting the weighting coefficients $\xi_i(t) \in \mathbb{R}^p$. Any type of functions can be included in Θ , with polynomials often serving as a suitable library for problems in computational mechanics [39]. It is also possible to assume that the functions in Θ have an explicit time dependency, thus getting $\mathbf{f}(\mathbf{x}(t), t) \approx \Xi^\top(t) \Theta^\top(\mathbf{x}(t), t)$, for example to treat non-autonomous (externally forced) systems [30].

Our goal is to estimate the values of the unknown coefficients Ξ and their time evolution from the assimilated data. It is important to note that this framework allows non-zero coefficients in Ξ to vanish or, conversely, zero-valued coefficients to increase over time. This flexibility enables potential changes in the composition of the functional terms that contribute to the system dynamics.

First, an initialisation of the coefficients Ξ is performed offline using static learning procedures on a limited training set of snapshots of the state vector trajectories. The training data are collected in the

following matrix:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}^\top(t_1) \\ \mathbf{x}^\top(t_2) \\ \vdots \\ \mathbf{x}^\top(t_T) \end{bmatrix} = \begin{bmatrix} x_1(t_1) & x_2(t_1) & \cdots & x_n(t_1) \\ x_1(t_2) & x_2(t_2) & \cdots & x_n(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ x_1(t_T) & x_2(t_T) & \cdots & x_n(t_T) \end{bmatrix}, \quad (6)$$

where $x_i(t_j)$ is the i -th entry of \mathbf{x} at the j -th time instant. Similarly, a matrix $\dot{\mathbf{X}} \in \mathbb{R}^{T \times n}$ is constructed by collecting the derivatives $\dot{\mathbf{x}}$. Thus, the functions of the library are applied to the rows of \mathbf{X} obtaining $\Theta(\mathbf{X}) \in \mathbb{R}^{T \times p}$ (see [28] for further details) and Eq. (5) translates at the levels of the training data as:

$$\dot{\mathbf{X}} \approx \Theta(\mathbf{X})\Xi. \quad (7)$$

Matrix Ξ is obtained by solving the following sparsity promoted least square regression problem:

$$\Xi = \arg \min_{\Xi'} \|\dot{\mathbf{X}} - \Theta(\mathbf{X})\Xi'\|_2 + \mathcal{L}(\Xi'), \quad (8)$$

where $\mathcal{L}(\Xi')$ is a sparsity promoting regulariser that could enforce, for example, a LASSO regression [40] or a Sequential Thresholded Least Squares (STLSQ) algorithm [28]. Notice that this initialization represents a static estimate, representing the best fitting constant coefficients over the set of training trajectories.

Next, we are interested to adapt \mathbf{f} to assimilate acquired data, in order to account for time-varying dynamics. To do so, we augment the state vector to include a subset of the SINDy weighting coefficients $\tilde{\xi}$, thus obtaining $\varkappa = [\mathbf{x}, \tilde{\xi}]^\top$. This means that the SINDy coefficients of this subset are treated as random variables. In other words, $\tilde{\xi}$ collects the entries Ξ that may evolve in the system, while the remaining coefficients are grouped into ξ_0 . Since many coefficients are usually set to zero through Eq. (8), we expect that many of them will remain unchanged. On the contrary, coefficients set to zero by SINDy may be included in $\tilde{\xi}$ to track potential emerging dynamic behaviours. Coefficients $\tilde{\xi}$ and ξ_0 are collected into $\tilde{\Xi}$ and $\Xi_0 \in \mathbb{R}^{p \times n}$, respectively, by multiplying Ξ through a Boolean matrix $\mathbf{B} \in \mathbb{R}^{p \times n}$ elementwise, that is by multiplying corresponding elements of \mathbf{B} and Ξ :

$$\Xi(t) = \tilde{\Xi}(t) + \Xi_0, \quad \text{with} \quad (9a)$$

$$\tilde{\Xi}(t) = \mathbf{B} \odot \Xi(t), \quad (9b)$$

$$\Xi_0 = (\mathbf{1} - \mathbf{B}) \odot \Xi(t), \quad (9c)$$

where \odot denotes the elementwise product operator and $\mathbf{1} \in \mathbb{R}^{p \times n}$ is a matrix whose entries are set to 1.

Consistently, the introduced notation allows the dynamic evolution of \mathbf{x} to be expressed as:

$$\mathbf{f}(\mathbf{x}(t)) \approx \Xi(t)^\top \Theta^\top(\mathbf{x}(t), t) = (\tilde{\Xi}(t) + \Xi_0)^\top \Theta^\top(\mathbf{x}(t)). \quad (10)$$

However, defining the augmented state \varkappa requires describing the dynamic evolution of $\tilde{\xi}$ as well. We assume that this evolution is driven by a white, zero mean stochastic process noise $\mathbf{q}_\xi \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_\xi)$ with diagonal covariance matrix $\mathbf{Q}_\xi \in \mathbb{R}^{e \times e}$. We thus introduce a unique process noise $\mathbf{q} = [\mathbf{q}_x, \mathbf{q}_\xi]^\top$, and a unique covariance matrix \mathbf{Q} collecting the entries of \mathbf{Q}_x and \mathbf{Q}_ξ . This means that the values of $\tilde{\xi}$ can be only modified in the correction stage on the basis of the mismatch between the incoming $\mathbf{y}(t_j)$ and the predicted $\mathbf{h}(\hat{\mathbf{x}}^-(t_j), t_j)$.

Ultimately, using SINDy to estimate \mathbf{f} enables to compute the Jacobian matrix $\mathbf{F}^+(t_{j-1})$ in a straightforward way as detailed in the following:

$$\mathbf{F}^+(t_{j-1}) = \begin{bmatrix} \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}^\top} \right|_{\hat{\varkappa}^+(t_{j-1})} & \left. \frac{\partial \mathbf{f}}{\partial \tilde{\xi}^\top} \right|_{\hat{\varkappa}^+(t_{j-1})} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} (\tilde{\Xi}^+(t_{j-1}) + \Xi_0)^\top \frac{\partial \Theta^\top(\mathbf{x})}{\partial \mathbf{x}^\top} \Big|_{\hat{\varkappa}^+(t_{j-1})} & \frac{\partial (\tilde{\Xi}(t_{j-1}))^\top}{\partial \tilde{\xi}} \tilde{\Theta}^\top(\mathbf{x}) \Big|_{\hat{\varkappa}^+(t_{j-1})} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad (11)$$

where it can be noted how the derivatives with respect to $\tilde{\xi}$ directly enters in the Jacobian definition. Implementation details of how the online adaption of the SINDy model is performed via the EKF by assimilating incoming measurements is reported in Algorithm 1.

Algorithm 1 Online adaptation of SINDy via EKF

Input: SINDy model $\mathbf{f}(\mathbf{x}(t)) \approx \mathbf{\Xi}(t)^\top \mathbf{\Theta}^\top (\mathbf{x}(t)) = (\tilde{\mathbf{\Xi}}(t) + \mathbf{\Xi}_0)^\top \mathbf{\Theta}^\top (\mathbf{x}(t))$; selecting operator $\mathbf{h}(\mathbf{x}(t_j), t_j)$; sequential measurements $\mathbf{y}(t_1), \dots, \mathbf{y}(t_T)$

Output: online update of the dynamic state variables \mathbf{x} and of the SINDy coefficients $\tilde{\mathbf{\xi}}$ through the estimates of $\hat{\mathbf{x}}^+(t_1), \dots, \hat{\mathbf{x}}^+(t_T)$, and $\hat{\mathbf{P}}^+(t_1), \dots, \hat{\mathbf{P}}^+(t_T)$.

Set: process and measurement noise covariance matrices \mathbf{Q} and \mathbf{R} ; the Boolean matrix \mathbf{B} .

- 1: $\mathbf{\Xi}(t) = \tilde{\mathbf{\Xi}}(t) + \mathbf{\Xi}_0$, with $\tilde{\mathbf{\Xi}}(t) = \mathbf{B} \odot \mathbf{\Xi}(t)$ and $\mathbf{\Xi}_0 = (\mathbf{I} - \mathbf{B}) \odot \mathbf{\Xi}(t)$
- 2: **for** $t_j = t_1, \dots, t_T$ **do**

Predictor phase (4th order Runge Kutta integration)

- 3: $\hat{\mathbf{x}}^-(t_j) = \hat{\mathbf{x}}^+(t_{j-1}) + \frac{\Delta t}{6}(\mathcal{K}_1^{\mathbf{x}} + 2\mathcal{K}_2^{\mathbf{x}} + 2\mathcal{K}_3^{\mathbf{x}} + \mathcal{K}_4^{\mathbf{x}})$
 - $\triangleright \mathcal{K}_1^{\mathbf{x}} = (\tilde{\mathbf{\Xi}}^+(t_{j-1}) + \mathbf{\Xi}_0)^\top \mathbf{\Theta}^\top (\hat{\mathbf{x}}^+(t_{j-1}))$
 - $\triangleright \mathcal{K}_2^{\mathbf{x}} = (\tilde{\mathbf{\Xi}}^+(t_{j-1}) + \mathbf{\Xi}_0)^\top \mathbf{\Theta}^\top (\hat{\mathbf{x}}^+(t_{j-1}) + \Delta t \frac{\mathcal{K}_1^{\mathbf{x}}}{2})$
 - $\triangleright \mathcal{K}_3^{\mathbf{x}} = (\tilde{\mathbf{\Xi}}^+(t_{j-1}) + \mathbf{\Xi}_0)^\top \mathbf{\Theta}^\top (\hat{\mathbf{x}}^+(t_{j-1}) + \Delta t \frac{\mathcal{K}_2^{\mathbf{x}}}{2})$
 - $\triangleright \mathcal{K}_4^{\mathbf{x}} = (\tilde{\mathbf{\Xi}}^+(t_{j-1}) + \mathbf{\Xi}_0)^\top \mathbf{\Theta}^\top (\hat{\mathbf{x}}^+(t_{j-1}) + \Delta t \mathcal{K}_3^{\mathbf{x}})$
- 4: $\hat{\mathbf{\xi}}^-(t_j) = \hat{\mathbf{\xi}}^+(t_{j-1})$
- 5: $\mathbf{P}^-(t_j) = \mathbf{P}^+(t_{j-1}) + \frac{\Delta t}{6}(\mathcal{K}_1^{\mathbf{P}} + 2\mathcal{K}_2^{\mathbf{P}} + 2\mathcal{K}_3^{\mathbf{P}} + \mathcal{K}_4^{\mathbf{P}})$
 - $\triangleright \mathcal{K}_1^{\mathbf{P}} = \mathbf{F}^+(t_{j-1})\mathbf{P}^+(t_{j-1}) + \mathbf{P}^+(t_{j-1})\mathbf{F}^{+\top}(t_{j-1}) + \mathbf{Q}_x$
 - $\triangleright \mathcal{K}_2^{\mathbf{P}} = \mathbf{F}^+(t_{j-1} + \frac{\Delta t}{2}) \left(\mathbf{P}^+(t_{j-1}) + \frac{\mathcal{K}_1^{\mathbf{P}}}{2} \right) + \left(\mathbf{P}^+(t_{j-1}) + \frac{\mathcal{K}_1^{\mathbf{P}}}{2} \right) \mathbf{F}^{+\top}(t_{j-1} + \frac{\Delta t}{2}) + \mathbf{Q}_x$
 - $\triangleright \mathcal{K}_3^{\mathbf{P}} = \mathbf{F}^+(t_{j-1} + \frac{\Delta t}{2}) \left(\mathbf{P}^+(t_{j-1}) + \frac{\mathcal{K}_2^{\mathbf{P}}}{2} \right) + \left(\mathbf{P}^+(t_{j-1}) + \frac{\mathcal{K}_2^{\mathbf{P}}}{2} \right) \mathbf{F}^{+\top}(t_{j-1} + \frac{\Delta t}{2}) + \mathbf{Q}_x$
 - $\triangleright \mathcal{K}_4^{\mathbf{P}} = \mathbf{F}^+(t_{j-1} + \frac{\Delta t}{2}) (\mathbf{P}^+(t_{j-1}) + \mathcal{K}_3^{\mathbf{P}}) + (\mathbf{P}^+(t_{j-1}) + \mathcal{K}_3^{\mathbf{P}}) \mathbf{F}^{+\top}(t_{j-1} + \Delta t) + \mathbf{Q}_x$
 - $\triangleright \mathbf{F}^+(t_{j-1}) = \left[\begin{array}{c} (\tilde{\mathbf{\Xi}}^+(t_{j-1}) + \mathbf{\Xi}_0)^\top \frac{\partial \mathbf{\Theta}^\top(\mathbf{x})}{\partial \mathbf{x}^\top} \Big|_{\hat{\mathbf{x}}^+(t_{j-1})} \quad \frac{\partial (\tilde{\mathbf{\Xi}}^+(t_{j-1}))^\top}{\partial \tilde{\mathbf{\xi}}} \tilde{\mathbf{\Theta}}^\top(\mathbf{x}) \Big|_{\hat{\mathbf{x}}^+(t_{j-1})} \\ \mathbf{0} \quad \mathbf{0} \end{array} \right]$

Corrector phase

- 6: $\mathbf{G}(t_j) = \mathbf{P}^-(t_j) \mathbf{H}^-(t_j)^\top (\mathbf{H}^-(t_j) \mathbf{P}^-(t_j) \mathbf{H}^-(t_j)^\top + \mathbf{R})^{-1}$
 - 7: $\hat{\mathbf{x}}^+(t_j) = \hat{\mathbf{x}}^-(t_j) + \mathbf{G}(t_j)(\mathbf{y}(t_j) - \mathbf{h}(\hat{\mathbf{x}}^-(t_j), t_j))$
 - $\triangleright \mathbf{H}^-(t_j) = \left[\begin{array}{c} \frac{\partial \mathbf{h}}{\partial \mathbf{x}^\top} \Big|_{\hat{\mathbf{x}}^-(t_j)} \quad \mathbf{0} \end{array} \right]$
 - 8: $\mathbf{P}^+(t_j) = (\mathbf{I} - \mathbf{G}(t_j) \mathbf{H}^-(t_j)) \mathbf{P}^-(t_j) (\mathbf{I} - \mathbf{G}(t_j) \mathbf{H}^-(t_j))^\top + \mathbf{G}(t_j) \mathbf{R} \mathbf{G}(t_j)^\top$
 - 9: **end for**
-

3. Numerical Results

The method outlined in Sec. 2 is now applied to three case studies. In Sec. 3.1, we examine a Lokta-Volterra model where all parameters evolve simultaneously, demonstrating our framework ability to construct and online update a SINDy model according to acquired measurements. In Sec. 3.2, we investigate a Selkov model to determine whether the adaptive SINDy model can track previously unseen dynamics after undergoing a bifurcation point controlled by an evolving parameter. All variables and parameters in these two case studies are treated as nondimensional. In Sec. 3.3, we assess the applicability of our approach to a pair of industry relevant engineering scenarios involving a MEMS device with a 1:2 internal resonance.

3.1. Lokta Volterra model

The Lokta-Volterra model is used to describe the dynamics of a biological system involving the interaction between two species, one acting as prey and the other as predator [41]. The system dynamics reads:

$$\dot{x}_1 = f_1(x_1, x_2) = ax_1 + bx_1x_2, \quad (12a)$$

$$\dot{x}_2 = f_2(x_1, x_2) = cx_2 + dx_1x_2, \quad (12b)$$

where x_1 is the prey population quantity, x_2 is the predator population quantity; $a \in \mathbb{R}^+$, $b \in \mathbb{R}^-$, $c \in \mathbb{R}^-$, $d \in \mathbb{R}^+$ are parameters characterising the interaction between the two species, and $\mathbf{f} = [f_1, f_2]^\top$.

We aim to evaluate whether the proposed procedure can track potential changes in the model parameters within the observation window $[0, T]$, with $T = 150$, by monitoring both x_1 and x_2 . To achieve this, we consider a function library Θ consisting of polynomial terms up to the second order. The corresponding coefficients Ξ are preliminary identified offline using STLSQ over a set of system trajectories not affected by noise, computed by numerical integration for the following model parameters $a = 1.0$, $b = -0.1$, $c = -1.5$ and $d = 0.075$. The STLSQ algorithm is performed with a regularisation strength $\delta_r = 0.05$, as default in the PySINDy package, and a threshold parameter $L = 5 \cdot 10^{-4}$. Specifically, δ_r determines the strength of the ℓ_2 regularisation promoting sparsity in the SINDy representation, while L fixes the threshold below which the entries of Ξ are set to 0. In the case of noisy data, noise tolerant versions of SINDy could be employed [42, 43]. As shown in Appendix A, SINDy successfully reconstructs the original model as can be noted by comparing the coefficients determined by SINDy with the ones that weight the relevant functional terms in the reference model used to train SINDy.

After the offline initialisation of the model, we employ the EKF to online assimilate noisy observations $\mathbf{y} = [\bar{x}_1, \bar{x}_2]^\top$ of the system, letting the parameters evolve according to Algorithm 1, and thereby consistently updating \mathbf{f} . This has involved incorporating a , b , c , d into $\tilde{\xi}$. The integration time step Δt , equal to the sampling time, has been set to $5.130 \cdot 10^{-3}$. The acquired measurements have been characterised by a Signal to Noise Ratio (SNR) of 25 with white noise generated numerically. To test the procedure in more challenging conditions, each parameter has been assigned a different evolution law. A sinusoidal variation, ranging from 0.8 to 1.2, has been considered for a , while a step change from -0.1 to -0.09 at $t = 50$ have been imposed on b ; c has been instead kept constant to assess the filter ability to maintain its estimate unchanged; lastly, d has been given a linear variation from 0.075 to 0.085. It is important to note that, by keeping unchanged the sign of the model coefficients, the stability portrait of the system remains unchanged, resulting in periodic trajectories which form closed orbits in the state-space surrounding the stable center $\mathbf{x}_{\text{eq}} = [\frac{d}{c}, \frac{a}{b}]^\top$.

The results of the identification are collected in Fig. 2 in terms both of tracking the dynamics of the system and of describing its parameter evolution. Target values are indicated with a bar, while noise-free version of the observations are labelled as \hat{x}_1 and \hat{x}_2 . The procedure has allowed us to successfully track the system dynamics even under evolving conditions in the predictive model \mathbf{f} , which had not been encountered during SINDy training. All parameters have been accurately and rapidly identified, particularly in response of the step change in b , proving how model adaptability can be attained by assimilating incoming measurements through EKF.

3.2. Selkov model

The Selkov model describes glycolysis by the following system of two ordinary differential equations [44]:

$$\dot{x}_1 = f_1(x_1, x_2) = \rho + \varsigma_1 x_1 - \vartheta_1 x_1 x_2^2, \quad (13a)$$

$$\dot{x}_2 = f_2(x_1, x_2) = \varsigma_2 x_1 + \psi x_2 + \vartheta_2 x_1 x_2^2, \quad (13b)$$

where x_1 and x_2 represent the concentration of two chemicals species in the glycolysis. Not all the coefficients that weight the functional terms usually vary when applying the Selkov model. Specifically, $\vartheta_1 = -1$, $\vartheta_2 = 1$, $\psi = -1$ are typically fixed. Similarly, the contribution of the cubic term $x_1 x_2^2$ is generally not parametrised, but it is included here with coefficients $\vartheta_1 \in \mathbb{R}^-$ and $\vartheta_2 \in \mathbb{R}^+$. Likewise, two separate parameters ς_1 and ς_2 have been used instead of the usual single coefficient ς , which would normally require changing the sign of the x_1 term in Eq. (13a)). This detailed parametrisation has been adopted for clarity, particularly to demonstrate whether SINDy, combined with the EKF, can estimate all these coefficients by observing x_1 and x_2 .

Particular attention is given to ρ , as varying its value can alter the system dynamics. By setting $\varsigma = 0.1$, the system undergoes two Hopf bifurcations [45] at $\rho \simeq 0.42$ and $\rho \simeq 0.79$. Specifically, for $\rho < 0.42$ and $\rho > 0.79$ the system has a stable fixed point, while for $0.42 < \rho < 0.79$ the fixed point loses stability, giving rise to a stable limit cycle. In the following, we show how the EKF is capable of detecting and tracking the system through the Hopf bifurcation occurring as ρ varies from 0.9 to 0.72, by assimilating noisy observations of x_1 and x_2 over the time window $[0, T = 300]$. This is accomplished even though the EKF is equipped with a SINDy model that has been trained offline on trajectories corresponding to a single fixed parameter $\rho = 0.92$.

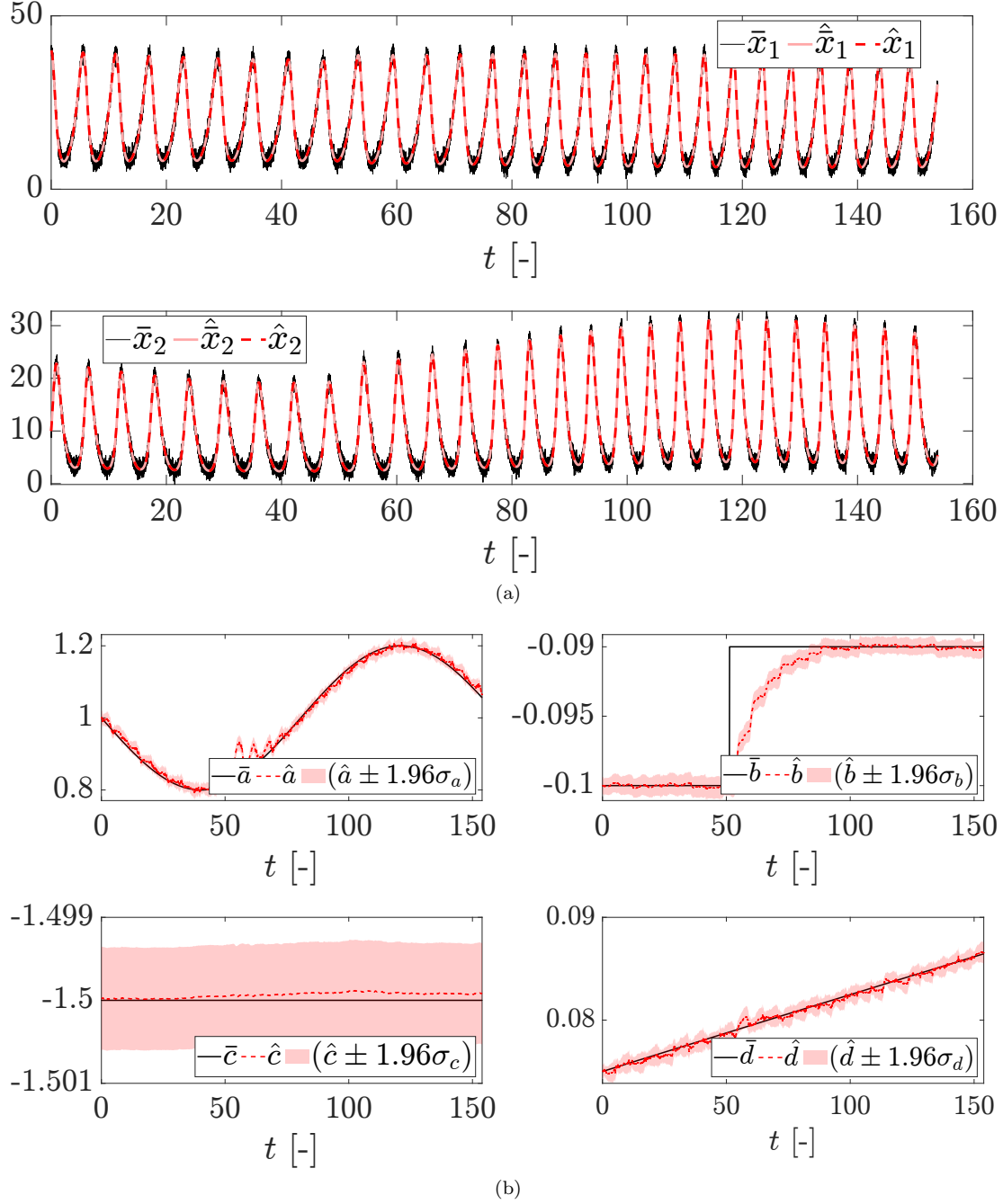


Figure 2: Lokta-Volterra model. In the top figure (a), the estimated states are plotted (red dotted line) against the assimilated noisy signals (black solid lines). Additionally, noise-free signals are reported in pink solid lines. In the bottom figure (b), the time-evolution of the estimated SINDy-model parameter are plotted (red dotted line) against the target time evolution (black solid line). The red shaded area represents 95% confidence interval of the estimates, determined using the posterior covariance.

Therefore, the predictive model is initialised on trajectories leading to a fixed point, being unaware of the parametric dependency and the corresponding bifurcating behavior. The time step for integrating Eq. 13 has been set to $\Delta t = 0.1$, which also serves as the sampling rate for the observations. Noisy observations have been generated by adding white noise to the signals, resulting in time series with a final SNR of 25.

SINDy has been initially trained using the SQTLSQ algorithm with $\delta_r = 0.05$ and $L = 5 \cdot 10^{-2}$, and a cubic polynomial functional library Θ . Results of the SINDy modelling are presented in Tab. A.2 in

Appendix A. In contrast to what is presented in Sec. 3.1, the function \mathbf{f} describing the evolution of system dynamics, as identified offline by SINDy, differs from Eq. (13) in both the values of the parameters that weight the functional terms and in the number of functional terms included. In particular, SINDy introduces a quadratic contribution x_1x_2 to model $\dot{x}_1 = f_1(x_1, x_2)$, weighting it with the parameter ι . It must be noted that increasing the SQTLSQ threshold to $L = 10^{-1}$ would prevent SINDy from the identification of this additional term. However, modifying L in this way would require a priori knowledge of the system dynamics, which is not always granted available. Assuming that this is one of those cases, we have investigated how the EKF performs when equipped with such, not correctly sparse, model. Here we have adopted an augmented state $\boldsymbol{\varkappa} = [\mathbf{x}^\top, \tilde{\boldsymbol{\xi}}^\top]^\top$, with $\mathbf{x} = [x_1, x_2]^\top$ and $\tilde{\boldsymbol{\xi}} = [\rho, \varsigma_1, \varsigma_2, \vartheta_1, \vartheta_2, \psi, \iota]$.

The filtering outcomes are presented in Figs. 3 and 4, which respectively illustrate the identification results for the variables tracking the system dynamics, x_1 and x_2 , and for the model parameters. In the second figure, it can be observed that the EKF is able not only to track the evolution of ρ but also to improve the estimate of the other parameters. Specifically, the filter sets $\iota = 0$, thereby restoring the correct sparsity of the system.

Notably, the recovery of the target parameter values has been achieved while tracking the system through two distinct dynamics regimes divided by a Hopf bifurcation. This is shown in Fig. 3a, which illustrates the evolution of x_1 and x_2 in $[0, T]$. Before the Hopf bifurcation at $t = 83.8$, the system approaches a stable fixed point, although the location of this point shifts due to variations in the parameter ρ . After the bifurcation, the system dynamics change, resulting in orbits forming a stable limit cycle. Interestingly, the trajectories used to train the SINDy model do not include limit cycle orbits. However, due to the coupling with the filter, the resulting data-driven model is able to generalise to a completely different dynamics. Such an outcome may be difficult to achieve with other ML techniques, highlighting the significant value of the proposed approach. In Fig. 3b, the previously discussed 3D representation is projected onto the (x_1, x_2) , (t, x_1) , and (t, x_2) planes for clarity.

3.3. MEMS arch

The proposed procedure is finally applied to a MEMS clamped shallow arch, whose geometry is shown in Fig. 5. The device is made by polycrystalline silicon with density equal to $D = 2330 \text{ kg/m}^3$, Young modulus $E = 1.670 \cdot 10^5 \text{ MPa}$ and Poisson ratio $\nu = 0.22$. A linear elastic behaviour has been assumed to model its constitutive behaviour.

This structure has been extensively studied due to its intriguing dynamic response, which is characterised by a 1:2 internal resonance, and the emergence of a quasi-periodic regime as a result of Neimark-Sacker (NS) bifurcations [46]. By adopting the two eigenmodes involved in the 1:2 resonance as master modes, a 2 Degrees of Freedom (DOFs) Reduced Order Model (ROM) was obtained in [38] via the static condensation approach [47]. In this method, a stress manifold is generated by applying to the arch static loadings proportional to the inertia of the master modes. This manifold was thus fitted with a third order polynomial to get an explicit expression for the nonlinear restoring forces. A representation of the eigenmodes involved in the arch resonance can be found in [38]. The two resonance frequencies are $\omega_1 = 0.43416 \text{ MHz}$ and $\omega_2 = 0.86367 \text{ MHz}$, respectively, while the Q factors are $Q_1 = 500$ and $Q_2 = 1000$. The connection between the Q factors and the damping coefficients will be specified in the following. The eigenmode amplitudes are represented by the modal coordinates u_1 and u_2 , measured in μm , while their time derivative are denoted by v_1 and v_2 . The following mass normalised ROM \mathbf{f} , here playing the role of physical model, expressed with $\tau = \omega_1 t$, is considered for the arch dynamics:

$$\dot{u}_1 = f_{1v}(v_1) = v_1, \quad (14a)$$

$$\dot{u}_2 = f_{2v}(v_2) = v_2, \quad (14b)$$

$$v_1 = f_1(u_1, u_2, v_1) = -k_1 u_1 - \mu_1 v_1 - a_{12} u_1 u_2 - b_{111} u_1^3 + B \cos(\Omega \tau), \quad (14c)$$

$$v_2 = f_2(u_1, u_2, v_2) = -k_2 u_2 - \mu_2 v_2 - a_{11} u_1^2 - b_{222} u_2^3, \quad (14d)$$

where: $\mu_1 = 1/Q_1$ and $\mu_2 = \omega_2/(Q_2 \omega_1)$ are the damping coefficients of the two modes; k_1 and k_2 are the linear stiffness coefficients; a_{12}, a_{11} are the parameters ruling the quadratic contributions to the elastic force; b_{111} and b_{222} are parameters ruling the cubic contributions; $B \cos(\Omega \tau)$ is the harmonic forcing amplified by the load amplifier $B = 0.0201 \mu\text{N} \mu\text{s}^2/\text{ng}$, that is applied by driving an alternated current through capacitor plates positioned near the arch.

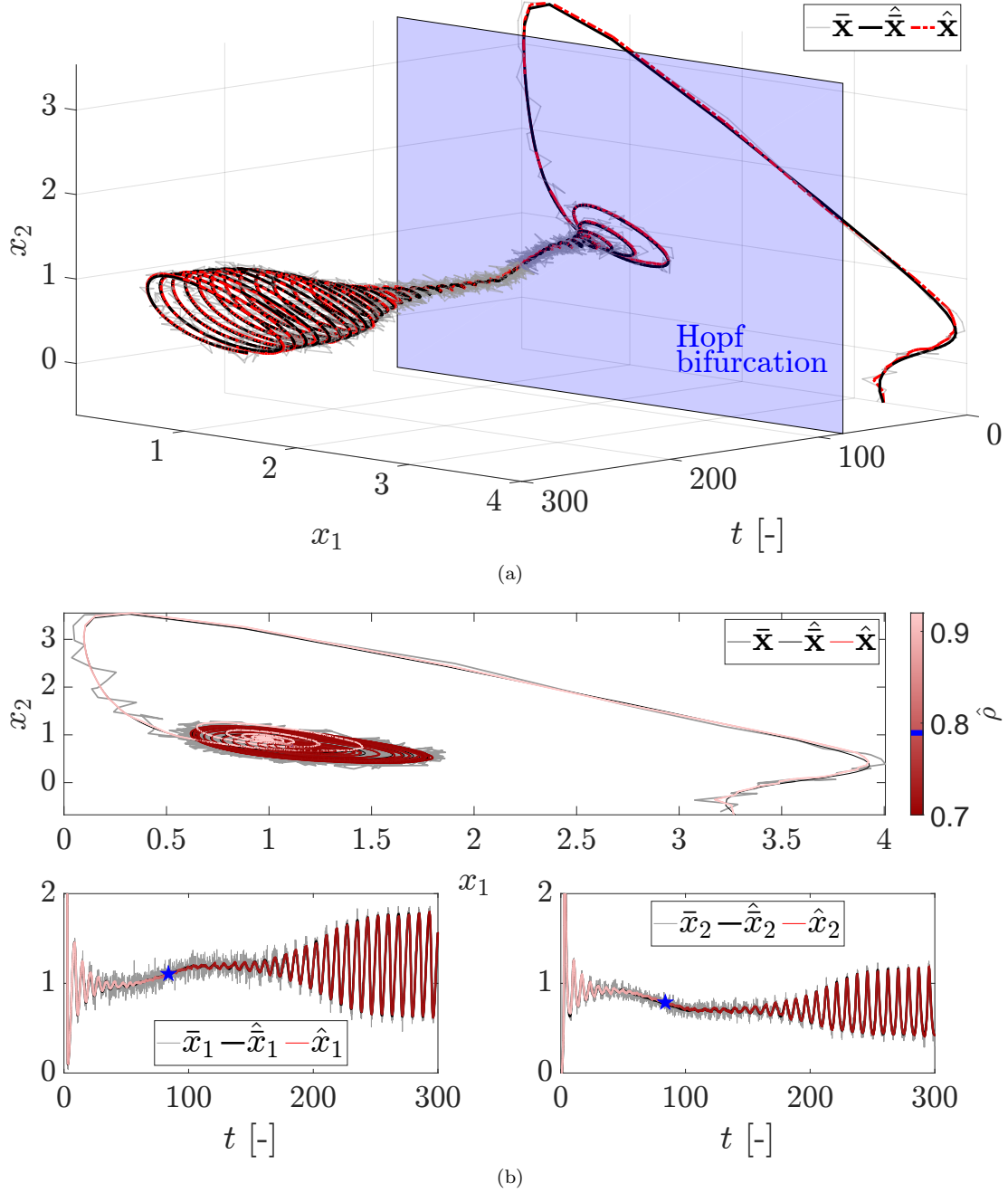
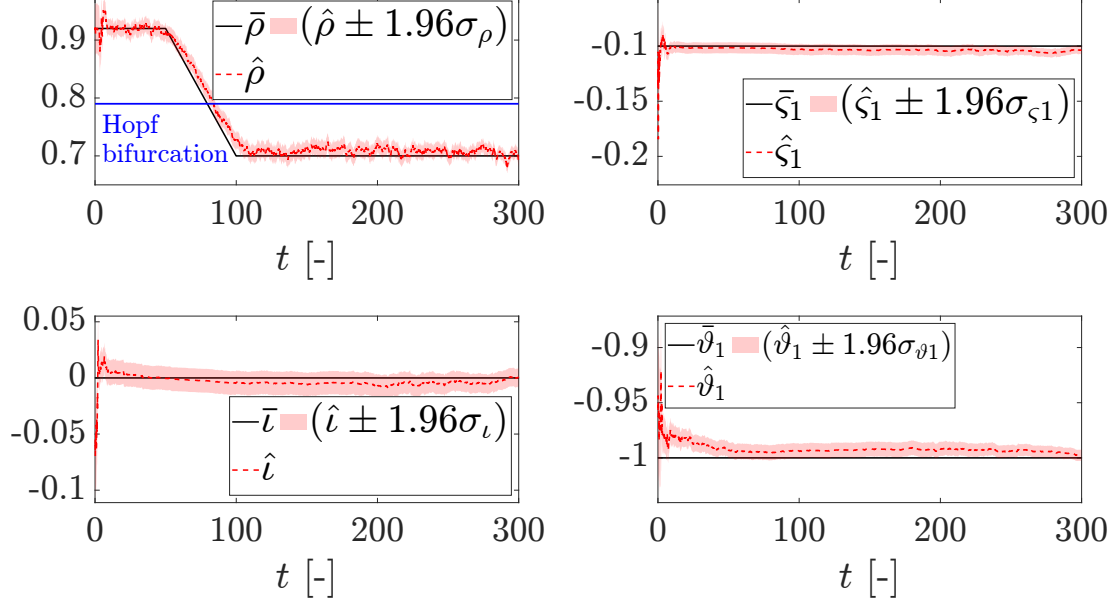
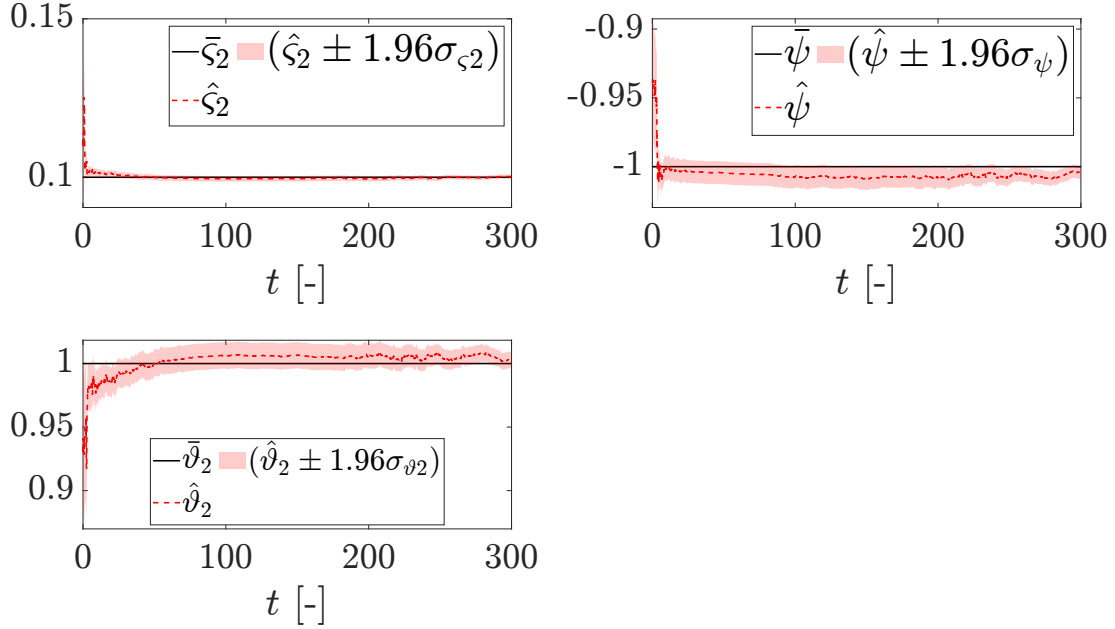


Figure 3: Selkov model. In the top figure (a), the estimated states are plotted (red dotted line) against the assimilated noisy signals (gray solid lines) using a 3D state-space-time representation. Additionally, noise-free signals are reported in black solid lines. In the bottom plot (b), the projection of this 3D representation onto the 2D planes is reported. Colouring depending on the estimate $\hat{\rho}$ of the parameter ruling the two types of dynamics of the system (stable focus and stable limit cycles). The blue colour (further marked with a star in the (t, x_1) and (t, x_2) planes) indicates when the system undergoes a Hopf bifurcation.

Compared to the original formulation adopted in [38], Eq.(14) retains the terms that govern the most significant aspects of the arch dynamics. Specifically, the internal resonance between the two modes is driven by the dependence of \dot{v}_2 on the quadratic term $a_{11}u_1^2$; consequently, we have also included $a_{12}u_1u_2$ to model \dot{v}_1 , as this term accordingly derives from the elastic potential whenever $a_{11}u_1^2$ is present in the formulation of \dot{v}_2 [47]. Cubic terms are also retained to capture the softening and hardening behaviour of the two peaks of the system Frequency Response Curve (FRC) shown in Fig. 6. The other terms of [38] are not



(a) f_1 parameter identification.



(b) f_2 parameter identification.

Figure 4: Selkov model. The time-evolution of the estimated SINDy-model parameter are plotted (red dotted line) against the target time evolution (black solid line). The top figure reports the evolution of the parameters entering $\dot{x}_1 = f_1(x_1, x_2)$; the bottom figure the evolution of the parameters entering $\dot{x}_2 = f_2(x_1, x_2)$. The red shaded area represents 95% confidence interval of the estimates, determined using the posterior covariance.

retained because they result only on a marginal shift of the bifurcation points. The FRC plots the maximum displacement \tilde{u}_1 of the first eigenmode at steady state against the non-dimensional excitation frequency Ω . The adopted formulation enables the onset of the 1:2 internal resonance, NS bifurcations, and the presence of frequency combs between these bifurcation points. The figure illustrates both the precise locations of the bifurcation points (on the right), and a comparison between the FRC obtained through Natural Continuation methods with that obtained from the time integration of Eq. (14) (on the left). The latter plot highlights the presence of frequency combs within the FRC region bounded by the NS bifurcation points, which are not

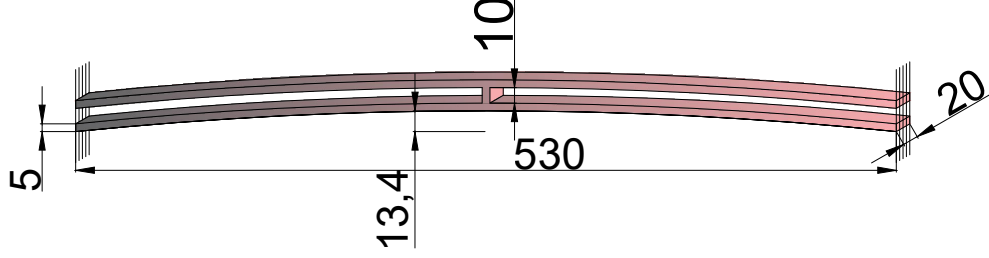


Figure 5: MEMS arch geometry. Dimensions in μm .

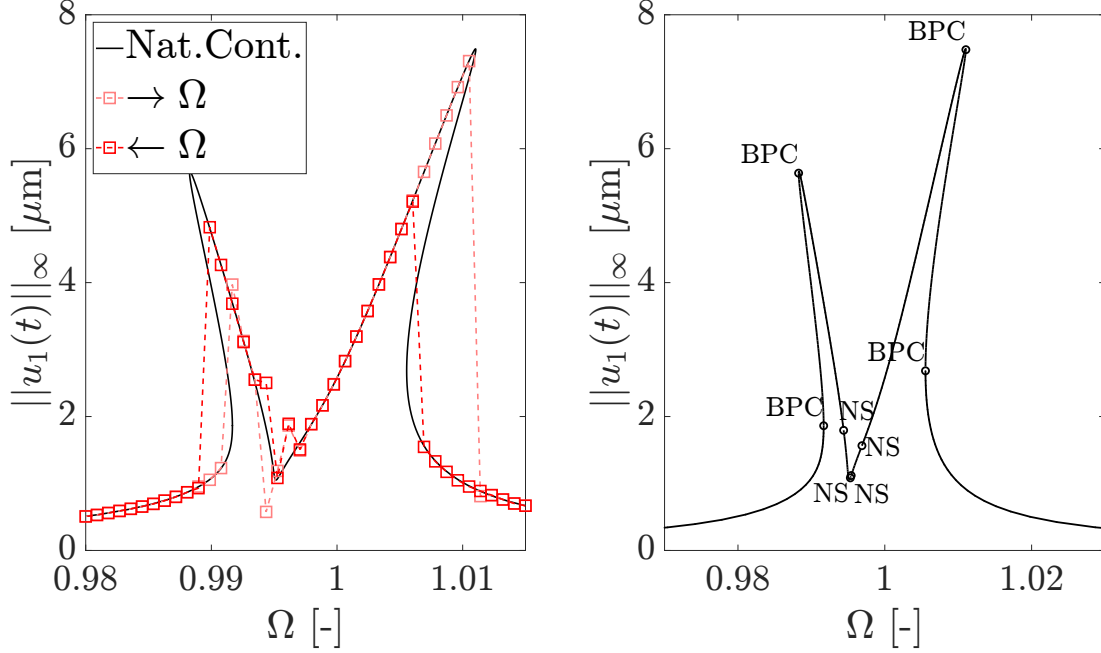


Figure 6: MEMS arch. On the left, comparison between the FRC obtained through numerical continuation (Nat. Cont.), and the FRCs obtained through time integration, both increasing ($\rightarrow \Omega$) and decreasing ($\leftarrow \Omega$) the excitation frequency Ω . On the right, the FRC of the system is reported highlighting the Branch Point Cycles (BPC) and Neimark-Sacker (NS) bifurcations.

observable when using a natural continuation method and assuming a single frequency periodic response. Natural continuation has been performed using *Matcont* [48].

First, we have recovered Eq. 14 via SINDy by observing the displacements and velocities of the arch eigenmodes, obtained by projecting the arch full field displacements and velocities. To this end, we have employed the SQTLSQ algorithm with a regularisation strength of $\delta_r = 0.05$ and threshold parameter $L = 10^{-5}$. For the composition of the function library Θ , polynomial terms up to the third order have been used in accordance with Eq. (14), together with a cosine term describing the forcing condition. The composition of Θ could have been inferred by developing the weak form of the dynamic equilibrium equation of the arch under large displacements and small strains [47]. Quadratic terms naturally arise from the asymmetric shape of the arch, while cubic terms emerge from the clamped-clamped condition [47, 46]. This type of reasoning, driven by the physics of the problem, may not be easily followed in other data-driven approaches to modelling dynamic systems such as recurrent neural networks, thus highlighting a significant advantage of the proposed procedure. The results of the identification are presented in the Appendix B by specifying the coefficients ruling the linear, quadratic and cubic terms. This shows that SINDy has successfully recovered the dynamics of the system although no parametric variability has been shown to SINDy. Indeed, the matrix \mathbf{X} collect trajectories for different forcing frequency Ω , while the linear, quadratic, and cubic terms have been fixed to the values considered in [38].

After applying SINDy, we have used the EKF to adapt the identified dynamics in two scenarios, each posing distinct challenges. Both scenarios have been treated numerically, with no noise sources taken into account. The first scenario (Sec. 3.3.1) involves updating the dynamic model \mathbf{f} , which describes the evolution of the two eigenmodes. Initially, this model is assumed to be identified via SINDy for a specific device, but it is then adapted to a MEMS with the same specifications however showing a different dynamic response. Since it is generally well assessed how to estimate the linear stiffness and damping terms, the focus has been on updating the estimate of the coefficients weighting the quadratic and cubic terms. Such an adaptation is commonly required in industrial settings to account for fabrication uncertainties, such as variations in the amount of overetch [49]. Another important application is sensor self calibration [50, 51]. Throughout its lifetime, a sensor may experience mechanical and thermal stress, that can cause unwanted shifts in its readings. Automatically compensating for errors is crucial to ensure the reliability of measurement systems, particularly in critical applications like structural health monitoring [52].

The second scenario (Sec. 3.3.2) involves recovering the coupling quadratic terms responsible for the internal resonance from data. This demonstrates how the procedure can enrich the dynamic description of the system while acquiring data. The only requirement is to include the relevant coefficients in the state vector, although initially set to an almost vanishing value. We believe that this scenario illustrates how the proposed approach could enable online identification of emerging contributions to the dynamics of a mechanical system.

3.3.1. Estimation of the quadratic and cubic nonlinearities

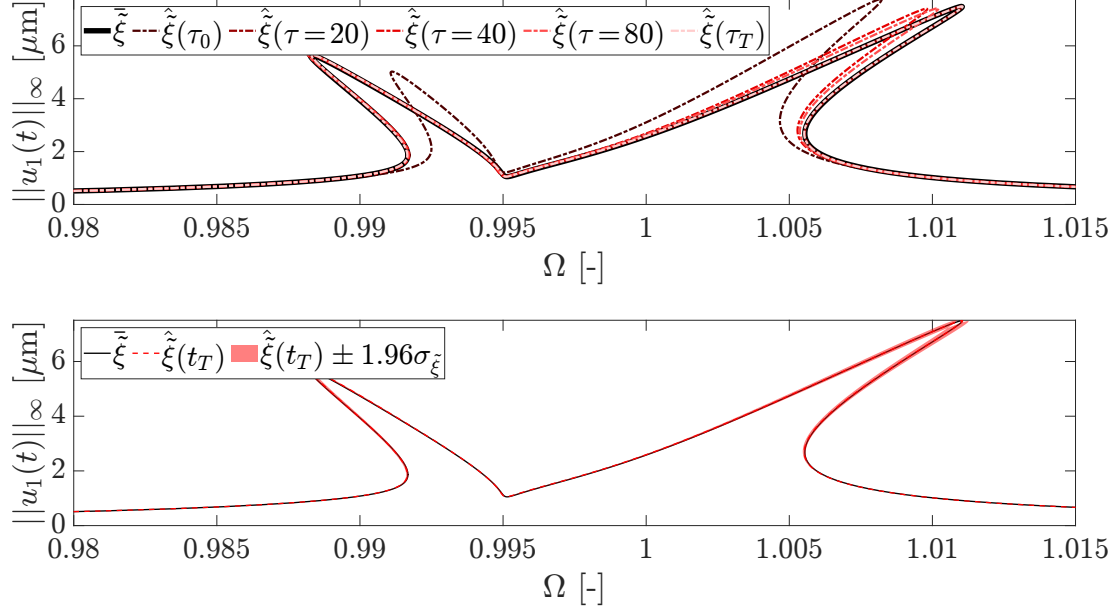
We consider adapting the values of the quadratic and cubic coefficients identified by SINDy (see Appendix B) to a different device which shares the same design but exhibits different dynamic behaviour which, for instance, is very typical of microstructures that are produced on identical wafers, but are characterized by slightly different geometrical features due to technological uncertainties. The adaptation is performed using the EKF, based on measurements of the eigenmode displacements and velocities. Specifically, this second device features quadratic and cubic coefficients that are 25% higher than those of the original device. The state vector $\mathbf{z} = [\mathbf{x}^\top, \boldsymbol{\zeta}_\Phi^\top]^\top$ includes both the dynamic variables $\mathbf{x}(t) = [u_1, u_2, v_1, v_2]^\top$ and the parameters $[a_{11}, a_{12}, b_{111}, b_{222}]^\top$, which are treated as time-dependant variables.

The EKF equipped with the modified SINDy model has been applied over a time window of $[\tau_0, \tau_T]$, with $\tau_0 = 0$ and $\tau_T = 5000$. The identification has been particularly challenging due to the small magnitude of the quadratic and cubic nonlinear coefficients. The system has been forced at $\Omega = 0.9899$, corresponding to the first peak of the FRC. This forcing frequency has been chosen to enhance sensitivity to the cubic contribution to the internal forces.

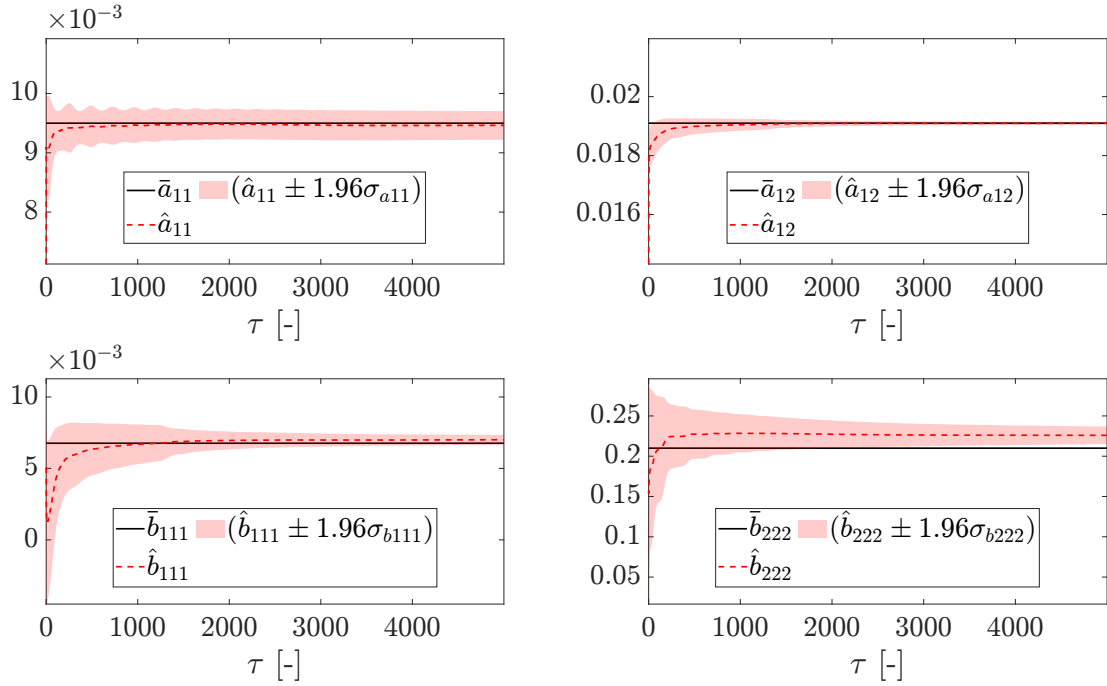
Results are shown in Fig. 7, showing excellent performance. Notably, key information about the dynamic response of the device are contained in the FRCs reported in Fig. 7a, where the adimensionalised excitation frequency Ω is plotted on the horizontal axis and \tilde{u}_1 on the vertical axis. Similar figures can be realised with \tilde{u}_2 on the vertical axis, leading to similar comments to those made for \tilde{u}_1 .

On top of Fig. 7a, it is shown how the FRC changes during the estimation. Specifically, it can be seen how far the FRC, initialised (at t_0) with the $\hat{\boldsymbol{\xi}}$ values determined by SINDy on the reference device, is from the target ones, and how quickly the EKF updates improve the estimations of $\hat{\boldsymbol{\xi}}$. A satisfactory outcome is achieved already at $\tau = 20$. This rapid response highlights the potential for online application of the proposed procedure, making it suitable for handling evolving environmental conditions that may affect the device dynamic response, or for responding to unexpected changes in the arch mechanics, such as damage from impacts. At the bottom of Fig. 7a, the estimated FRC is shown along with its corresponding uncertainty bounds. These bounds are generally very tight but become slightly larger close to the FRC peaks. This widening is more pronounced near the peak corresponding at $\Omega = 1.012$, likely due to the chosen excitation frequency being related to the first peak.

Fig. 7b illustrates the time evolution of the parameter estimation for $\hat{\boldsymbol{\xi}}$. It can be noted that the estimated \hat{b}_{222} does not exactly match the target value of b_{222} . However, we argue that the filter is unlikely to improve further, as the system dynamics prediction, as shown by the FRCs, is already highly accurate. In contrast, the estimation of the other cubic term coefficient \hat{b}_{111} , as well as the of quadratic terms, is excellent, with a remarkable reduction in estimation uncertainties over the course of the analysis.



(a) FRC modification during the identification (on top); FRC with uncertainty bounds at $t = T$ (on bottom).



(b) Parameter identification.

Figure 7: MEMS arch, estimation of the quadratic and cubic nonlinearities. In the top figure, the FRCs corresponding to the models estimated during the identification (red palette dotted lines) are compared with the target FRC (black solid line). In the central figure, The FRC identified at the end of the analysis (red dotted line) is plot against the target FRC (black solid line), with the red shaded area representing the 95% confidence intervals of the $\hat{\xi}$ estimates, determined using the posterior covariance. In the bottom figure, the time-evolution of the estimated SINDy-model parameter are plotted (red dotted line) against the target time evolution (black solid line).

3.3.2. Discovering the internal resonance

We now consider the case in which a_{11} is set to small enough ($1 \cdot 10^{-9}$) such that it does not induce internal resonance in the arch. It is noted that a_{11} can not be set exactly to zero, as this would eliminate

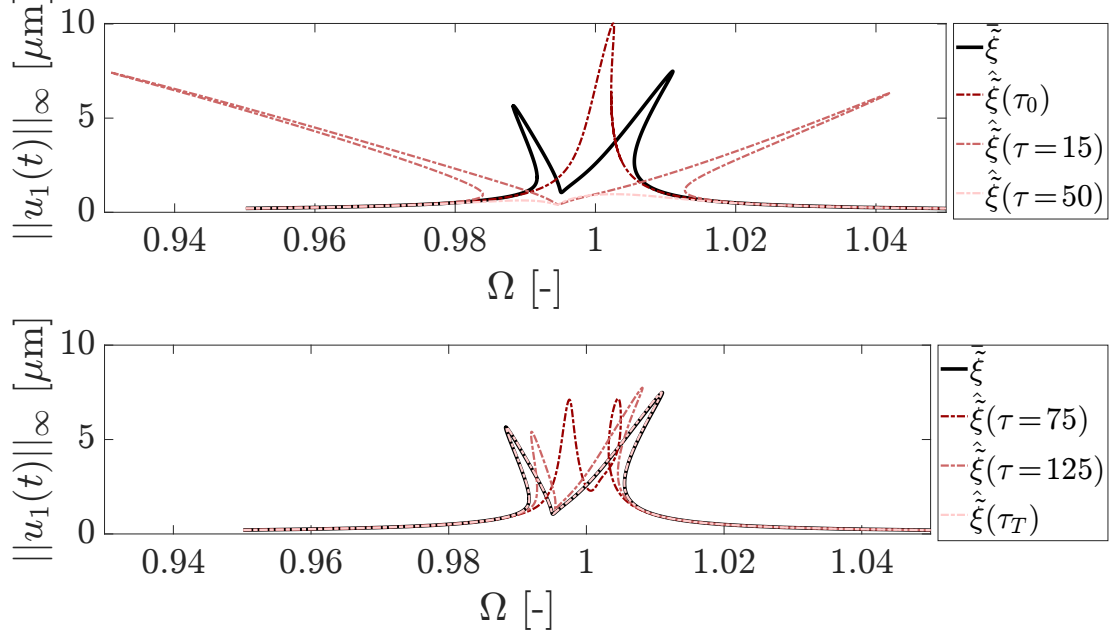


Figure 8: MEMS arch, discovering the internal resonance. The FRCs corresponding to the models estimated during the identification (red palette dotted lines) are compared with the target FRC (black solid line).

sensitivity to the parameter when computing the Jacobian matrix. We have thus included a_{11} in the state vector $\mathbf{x} = [\mathbf{x}, \tilde{\xi}]$ among the coefficients $\tilde{\xi} = [a_{11}, a_{12}]^\top$. Unlike the previous discussion, we have assumed that b_{111} and b_{222} are known to simplify the system identification, while we have still considered to underestimate the value of a_{12} by 25%.

Results are shown in Fig. 8. At $t = 0$, the FRC of the system, based on the initial estimates $\tilde{\xi}(\tau = 0)$, exhibits a single peak, as the value of a_{11} is too small to induce the internal resonance. As the quadratic terms estimates are updated and the corresponding FRCs are plotted, the system dynamics gradually converge to the true behaviour. Specifically, at $\tau = 15$, the initial peak levels out, and the appearance of two small amplified hills indicates that the identified system exhibits some interaction between the two eigenmodes. By $\tau = 50$, the internal resonance becomes marked, with two peaks showing significant hardening and softening behaviours. At $\tau = 125$, the identified system has the correct FRC shape, thus almost perfectly matching the target $\tilde{\xi}_\Phi$ at the end of the analysis (τ_T).

4. Conclusions

A combined approach exploiting the extended Kalman Filter (EKF) and the Sparse Identification of Nonlinear Dynamics (SINDy) technique has been adopted for online data assimilation and adaptation of nonlinear dynamics systems. SINDy is first used in a preliminary stage to identify a dynamic model, avoiding biases from incorrect assumptions. Second, the EKF is employed to data assimilation, directly updating the SINDy parameters. This procedure has shown the ability to track time-evolving systems, performing joint state-parameter estimation even for SINDy coefficients not accounted for during the relevant initial SINDy training. Notably, the EKF has been able to track systems that have undergone unseen bifurcations in their dynamics behaviour.

The promising results suggest that this procedure could be applied to a wide range of time-varying problems in dynamics. Future work could extend the method to partially unobserved system by incorporating time-delay embedding techniques [53], similarly to [30]. Conversely, in cases where the number of observations exceeds the number of dominant dynamic modes, autoencoders architectures could be employed. To further enhance the robustness of this approach, automatic procedures for the EKF tuning will be also addressed in future research.

Code and data accessibility

The source code of the proposed method is made available from the GitHub repository: <https://github.com/ContiPaolo/EKF-SINDy> [54].

Acknowledgments

The authors would like to thank Alessio Colombo (Politecnico di Milano) for the invaluable insights on numerical continuation methods. LR is supported by the Joint Research Platform “Sensor sysTEms and Advanced Materials” (STEAM) between Politecnico di Milano and STMicroelectronics. AM acknowledges the project FAIR (Future Artificial Intelligence Research), funded by the NextGenerationEU program within the PNRR-PE-AI scheme (M4C2, Investment 1.3, Line on Artificial Intelligence). AM and PC acknowledge the PRIN 2022 Project “Numerical approximation of uncertainty quantification problems for PDEs by multi-fidelity methods (UQ-FLY)” (No. 202222PACR), funded by the European Union - NextGenerationEU. AF acknowledges the PRIN 2022 Project “DIMIN- DIgital twins of nonlinear MIcrostructures with iNnovative model-order-reduction strategies” (No. 2022XATLT2) funded by the European Union - NextGenerationEU.

Appendix A.

In the following, we report the comparison between the (reference) model parameters used for the training of SINDy, weighting the relevant functional terms, and the parameters identified by SINDy for the Lokta-Volterra model (Tab. A.1), the Selkov model (Tab. A.2), and the MEMS arch (Tab. A.3).

Lokta Volterra model.

		$\dot{x}_1 = f_1(x_1, x_2)$		$\dot{x}_2 = f_2(x_1, x_2)$	
Coefficients	Relevant term	Reference	SINDy	Reference	SINDy
a	x_1	1.000	1.000	0	0
b	$x_1 x_2$	-0.1000	-0.1000	0	0
c	x_2	0	0	-1.500	-1.500
d	$x_1 x_2$	0	0	$7.500 \cdot 10^{-2}$	$7.500 \cdot 10^{-2}$

Table A.1: Lokta Volterra model. The “Reference” columns report the parameters used for the training of SINDy; in the “SINDy” columns, we list the parameter values identified by SINDy.

Selkov model.

		$\dot{x}_1 = f_1(x_1, x_2)$		$\dot{x}_2 = f_2(x_1, x_2)$	
Coefficients	Relevant term	Reference	SINDy	Reference	SINDy
ρ	1.0	0.9200	0.9234	0	0
ς_1	x_1	-0.1000	$-9.389 \cdot 10^{-2}$	0	0
ς_2	x_1	0	0	0.1000	0.1082
ψ	x_2	0	0	-1.000	-0.9343
ι	$x_1 x_2$	0	$-7.641 \cdot 10^{-2}$	0	0
ϑ_1	$x_1 x_2^2$	-1.000	-0.9294	0	0
ϑ_2	$x_1 x_2^2$	0	0	1.000	0.9185

Table A.2: Selkov model. The “Reference” columns report the parameters used for the training of SINDy; in the “SINDy” columns, we list the parameter values identified by SINDy.

$\dot{v}_1 = f_1(u_1, u_2, v_1)$				$\dot{v}_2 = f_2(u_1, u_2, v_2)$	
Coefficients	Relevant term	Reference	SINDy	Reference	SINDy
k_1	u_1	1.000	1.000	0	0
k_2	u_2	0	0	3.957	3.957
μ_1	v_1	$2.000 \cdot 10^{-3}$	$2.000 \cdot 10^{-3}$	0	0
μ_2	v_2	0	0	$2.000 \cdot 10^{-3}$	$1.999 \cdot 10^{-3}$
a_{11}	u_1^2	0	0	$7.125 \cdot 10^{-3}$	$7.124 \cdot 10^{-3}$
a_{12}	$u_1 u_2$	$1.433 \cdot 10^{-2}$	$1.432 \cdot 10^{-2}$	0	0
b_{111}	u_1^3	$5.074 \cdot 10^{-5}$	$5.080 \cdot 10^{-5}$	0	0
b_{222}	u_2^3	0	0	$1.575 \cdot 10^{-3}$	$1.575 \cdot 10^{-3}$
-	$B \cos(\Omega \tau)$	1.000	0.9999	0	0

Table A.3: MEMS arch. Coefficients weighting the linear, quadratic and cubic contribution to the elastic force. The “Reference” column reports the reduced order model coefficients determined in [38] and used for the training of SINDy; in the “SINDy” columns, we list the parameter values identified by SINDy.

Appendix B.

Lokta Volterra	
x_1, x_2	10^{-3}
a	10^{-4}
b, c, d	10^{-7}
Selkov	
x_1, x_2	10^{-8}
$\rho, \varsigma_1, \vartheta_1, \vartheta_2, \psi, \iota$	10^{-3}
ς_2	10^{-4}
MEMS arch	
Quadratic and cubic nonlinearities	
u_1, u_2, v_1, v_2	10^{-8}
a_{11}	$5 \cdot 10^{-4}$
a_{12}	10^{-4}
b_{111}	10^{-5}
b_{222}	$5 \cdot 10^{-3}$
Discovering internal resonance	
u_1, u_2, v_1, v_2	10^{-7}
a_{11}, a_{12}	10^{-4}

Table B.4: Initial covariance $\mathbf{P}(t_0)$, diagonal terms. In the left column, we specify the augmented state quantities to which these values are related.

Lokta Volterra	
x_1, x_2	10^{-3}
a	$5 \cdot 10^{-5}$
b	10^{-8}
c	10^{-14}
d	$8 \cdot 10^{-8}$
Selkov	
x_1, x_2	10^{-6}
$\rho, \varsigma_1, \varsigma_2$	10^{-14}
$\vartheta_1, \vartheta_2, \psi, \iota$	10^{-12}
MEMS arch	
Quadratic and cubic nonlinearities	
u_1, u_2	10^{-7}
v_1	10^{-5}
v_2	$8 \cdot 10^{-6}$
a_{11}	10^{-7}
a_{12}	10^{-12}
b_{111}	10^{-14}
b_{222}	10^{-16}
Discovering internal resonance	
u_1, u_2	10^{-8}
v_1, v_2	10^{-5}
a_{11}	10^{-7}
a_{12}	10^{-9}

Table B.5: Process noise \mathbf{Q} , diagonal terms. In the left column, we specify the augmented state quantities to which these values are related.

Lokta Volterra	
x_1, x_2	1.0
Selkov	
x_1, x_2	$5 \cdot 10^{-4}$
MEMS arch	
Quadratic and cubic nonlinearities	
u_1, u_2	10^{-1}
v_1, v_2	10^{-3}
Discovering internal resonance	
u_1, u_2	10^{-1}
v_1, v_2	10^{-3}

Table B.6: Measurement noise \mathbf{R} , diagonal terms. In the left column, we specify the observed quantities to which these values are related.

References

- [1] C. Farrar, K. Worden, Structural Health Monitoring: A Machine Learning Perspective, Wiley, Hoboken, NJ, 2013. doi:10.1002/9781118443118.

- [2] N. Fonzi, S. L. Brunton, U. Fasel, Data-driven nonlinear aeroelastic models of morphing wings for control, *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 476 (2239) (2020) 20200079. doi:10.1098/rspa.2020.0079.
- [3] A. Corigliano, R. Ardito, C. Comi, A. Frangi, A. Ghisi, S. Mariani, *Accelerometers*, John Wiley & Sons, Ltd, 2018, Ch. 4, pp. 91–108. doi:10.1002/9781119053828.ch4.
- [4] D. J. Wagg, K. Worden, R. J. Barthorpe, P. Gardner, Digital Twins: State-of-the-Art and Future Directions for Modeling and Simulation in Engineering Dynamics Applications, *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering* 6 (3) (2020) 030901. doi:10.1115/1.4046739.
- [5] M. Torzoni, M. Tezzele, S. Mariani, A. Manzoni, K. E. Willcox, A digital twin framework for civil engineering structures, *Computer Methods in Applied Mechanics and Engineering* 418 (2024) 116584. doi:10.1016/j.cma.2023.116584.
- [6] M. G. Kapteyn, J. V. R. Pretorius, K. E. Willcox, A probabilistic graphical model foundation for enabling predictive digital twins at scale, *Nature Computational Science* 1 (2021) 337–347. doi:10.1038/s43588-021-00069-0.
- [7] R. E. Kalman, A new approach to linear filtering and prediction problems, *Journal of Basic Engineering* 82 (1) (1960) 35–45. doi:10.1115/1.3662552.
- [8] D. Simon, *The discrete-time Kalman filter*, John Wiley & Sons, Ltd, 2006, Ch. 5, pp. 121–148. doi:10.1002/0470045345.ch5.
- [9] E. Wan, 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 (Cat. No.00EX373)*, 2000, pp. 153–158. doi:10.1109/ASSPCC.2000.882463.
- [10] S. Mariani, A. Ghisi, Unscented Kalman filtering for nonlinear structural dynamics, *Nonlinear Dynamics* 49 (2007) 131–150. doi:10.1007/s11071-006-9118-9.
- [11] L. Rosafalco, S. E. Azam, S. Mariani, A. Corigliano, System identification via unscented Kalman filtering and model class selection, *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering* 10 (1) (2024) 04023063. doi:10.1061/AJRUA6.RUENG-1085.
- [12] A. Nóvoa, A. Racca, L. Magri, Inferring unknown unknowns: Regularized bias-aware ensemble Kalman filter, *Computer Methods in Applied Mechanics and Engineering* 418 (2024) 116502. doi:10.1016/j.cma.2023.116502.
- [13] M. Song, R. Astroza, H. Ebrahimian, B. Moaveni, C. Papadimitriou, Adaptive kalman filters for nonlinear finite element model updating, *Mechanical Systems and Signal Processing* 143 (2020) 106837. doi:10.1016/j.ymssp.2020.106837.
- [14] S. Greś, M. Döhler, V. K. Dertimanis, E. N. Chatzi, Subspace-based noise covariance estimation for kalman filter in virtual sensing applications, *Mechanical Systems and Signal Processing* 222 (2025) 111772. doi:10.1016/j.ymssp.2024.111772.
- [15] N. Bilgin, A. Olivier, Joint bayesian estimation of process and measurement noise statistics in nonlinear kalman filtering, *Mechanical Systems and Signal Processing* 223 (2025) 111836. doi:10.1016/j.ymssp.2024.111836.
- [16] S. Cuomo, M. De Rosa, F. Piccialli, L. Pompameo, Railway safety through predictive vertical displacement analysis using the pinn-ekf synergy, *Mathematics and Computers in Simulation* 223 (2024) 368–379. doi:10.1016/j.matcom.2024.04.026.
- [17] M. Impraimakis, Deep recurrent-convolutional neural network learning and physics kalman filtering comparison in dynamic load identification, *Structural Health Monitoring* 0 (0) (0) 14759217241262972. doi:10.1177/14759217241262972.

- [18] F. Gatti, L. Rosafalco, G. Colombero, S. Mariani, A. Corigliano, Multi-storey shear type buildings under earthquake loading: Adversarial learning-based prediction of the transient dynamics and damage classification, *Soil Dynamics and Earthquake Engineering* 173 (2023) 108141. doi:10.1016/j.soildyn.2023.108141.
- [19] H. Joseph, G. Quaranta, B. Carboni, W. Lacarbonara, Deep learning architectures for data-driven damage detection in nonlinear dynamic systems under random vibrations, *Nonlinear Dynamics* 223 (2024). doi:10.1007/s11071-024-10270-1.
- [20] H. Coskun, F. Achilles, R. DiPietro, N. Navab, F. Tombari, Long short-term memory Kalman filters: Recurrent neural estimators for pose regularization, in: *2017 IEEE International Conference on Computer Vision (ICCV)*, October 22-29, Venezia, Italy, 2017, pp. 5525–5533. doi:10.1109/ICCV.2017.589.
- [21] W. Liu, Z. Lai, K. Bacsá, E. Chatzi, Neural extended Kalman filters for learning and predicting dynamics of structural systems, *Structural Health Monitoring* 23 (2) (2024) 1037–1052. doi:10.1177/14759217231179912.
- [22] A. Olivier, M. D. Shields, L. Graham-Brady, Bayesian neural networks for uncertainty quantification in data-driven materials modeling, *Computer Methods in Applied Mechanics and Engineering* 386 (2021) 114079. doi:10.1016/j.cma.2021.114079.
- [23] A. Cicirello, Physics-enhanced machine learning: a position paper for dynamical systems investigations (2024). arXiv:2405.05987.
URL <https://arxiv.org/abs/2405.05987>
- [24] L. Rosafalco, A. Manzoni, S. Mariani, A. Corigliano, Combined Model Order Reduction Techniques and Artificial Neural Network for Data Assimilation and Damage Detection in Structures, Springer International Publishing, Cham, 2022, pp. 247–259. doi:10.1007/978-3-030-70787-3_16.
- [25] P. Seventekidis, D. Giagopoulos, Model error effects in supervised damage identification of structures with numerically trained classifiers, *Mechanical Systems and Signal Processing* 184 (2023) 109741. doi:10.1016/j.ymssp.2022.109741.
- [26] P. Conti, G. Gobat, S. Fresca, A. Manzoni, A. Frangi, Reduced order modeling of parametrized systems through autoencoders and SINDy approach: continuation of periodic solutions, *Computer Methods in Applied Mechanics and Engineering* 411 (2023) 116072. doi:<https://doi.org/10.1016/j.cma.2023.116072>.
- [27] A. Opreni, A. Vizzaccaro, C. Touzé, A. Frangi, High-order direct parametrisation of invariant manifolds for model order reduction of finite element structures: application to generic forcing terms and parametrically excited systems, *Nonlinear Dynamics* 111 (2023) 5401–5447. doi:10.1007/s11071-022-07978-3.
- [28] S. L. Brunton, J. L. Proctor, J. N. Kutz, Discovering governing equations from data by sparse identification of nonlinear dynamical systems, *Proceedings of the National Academy of Sciences* 113 (15) (2016) 3932–3937. doi:10.1073/pnas.1517384113.
- [29] C. Lathourakis, A. Cicirello, Physics enhanced sparse identification of dynamical systems with discontinuous nonlinearities, *Nonlinear Dynamics* 112 (2024) 11237–11264. doi:10.1007/s11071-024-09652-2.
- [30] L. Rosafalco, P. Conti, A. Manzoni, S. Mariani, A. Frangi, EKF-SINDy: Empowering the extended kalman filter with sparse identification of nonlinear dynamics, *Computer Methods in Applied Mechanics and Engineering* 431 (2024) 117264. doi:<https://doi.org/10.1016/j.cma.2024.117264>.
- [31] T. Dardeno, K. Worden, N. Dervilis, R. Mills, L. Bull, On the hierarchical bayesian modelling of frequency response functions, *Mechanical Systems and Signal Processing* 208 (2024) 111072. doi:10.1016/j.ymssp.2023.111072.
- [32] S. Li, Y. Yang, Data-driven modeling of bifurcation systems by learning the bifurcation parameter generalization, *Nonlinear Dynamics* 415 (2023) 116204. doi:10.1007/s11071-024-10304-8.

- [33] R.-S. Götte, J. Timmermann, Estimating states and model uncertainties jointly by a sparsity promoting UKF, *IFAC-PapersOnLine* 56 (1) (2023) 85–90, 12th IFAC Symposium on Nonlinear Control Systems NOLCOS 2022. doi:10.1016/j.ifacol.2023.02.015.
- [34] A. Pal, S. Nagarajaiah, Sparsity promoting algorithm for identification of nonlinear dynamic system based on unscented kalman filter using novel selective thresholding and penalty-based model selection, *Mechanical Systems and Signal Processing* 212 (2024) 111301. doi:10.1016/j.ymssp.2024.111301.
- [35] S. J. Julier, J. J. LaViola, On kalman filtering with nonlinear equality constraints, *IEEE Transactions on Signal Processing* 55 (6) (2007) 2774–2784. doi:10.1109/TSP.2007.893949.
- [36] J. M. Stevens-Haas, Y. Bhangale, J. Nathan Kutz, A. Aravkin, Learning nonlinear dynamics using kalman smoothing, *IEEE Access* 12 (2024) 138564–138574. doi:10.1109/ACCESS.2024.3465390.
- [37] J. Wang, J. Moreira, Y. Cao, B. Gopaluni, Time-variant digital twin modeling through the Kalman-generalized sparse identification of nonlinear dynamics, in: *2022 American Control Conference (ACC)*, June 8-10, Atlanta, Georgia, USA, 2022, pp. 5217–5222. doi:10.23919/ACC53348.2022.9867786.
- [38] G. Gobat, V. Zega, P. Fedeli, L. Guerinoni, C. Touzé, A. Frangi, Reduced order modelling and experimental validation of a MEMS gyroscope test-structure exhibiting 1:2 internal resonance, *Scientific Reports* 11 (2021) 16390. doi:10.1038/s41598-021-95793-y.
- [39] K. P. Champion, S. L. Brunton, J. N. Kutz, Discovery of nonlinear multiscale systems: Sampling strategies and embeddings, *SIAM Journal on Applied Dynamical Systems* 18 (1) (2019) 312–333. doi:10.1137/18M1188227.
- [40] T. Hastie, R. Tibshirani, M. Wainwright, *Statistical learning with sparsity*, Monographs on statistics and applied probability 143 (143) (2015) 8.
- [41] A. J. Lotka, Analytical note on certain rhythmic relations in organic systems, *Proceedings of the National Academy of Sciences* 6 (7) (1920) 410–415. doi:10.1073/pnas.6.7.410.
- [42] U. Fasel, J. N. Kutz, B. W. Brunton, S. L. Brunton, Ensemble-SINDy: Robust sparse model discovery in the low-data, high-noise limit, with active learning and control, *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 478 (2260) (2022) 20210904. doi:10.1098/rspa.2021.0904.
- [43] P. Conti, J. Kneifl, A. Manzoni, A. Frangi, J. Fehr, S. L. Brunton, J. N. Kutz, VENI, VINDy, VICI: a variational reduced-order modeling framework with uncertainty quantification (2024). arXiv:2405.20905. URL <https://arxiv.org/abs/2405.20905>
- [44] E. E. Sel'kov, Self-oscillations in glycolysis 1. a simple kinetic model, *European Journal of Biochemistry* 4 (1) (1968) 79–86. doi:10.1111/j.1432-1033.1968.tb00175.x.
- [45] R. Muñoz-Alicea, Introduction to bifurcations and the hopf bifurcation theorem for planar systems, *Operation for Mathematics* 640 (11) (2011).
- [46] G. Gobat, V. Zega, P. Fedeli, C. Touzé, A. Frangi, Frequency combs in a mems resonator featuring 1:2 internal resonance: ab initio reduced order modelling and experimental validation, *Nonlinear dynamics* 111 (2023) 2991–3017. doi:10.1007/s11071-022-08029-7.
- [47] A. Frangi, G. Gobat, Reduced order modelling of the non-linear stiffness in mems resonators, *International Journal of Non-Linear Mechanics* 116 (2019) 211–218. doi:10.1016/j.ijnonlinmec.2019.07.002.
- [48] A. Dhooge, W. Govaerts, Y. A. Kuznetsov, MATCONT: A MATLAB package for numerical bifurcation analysis of ODEs, *ACM Transactions on Mathematical Software* 29 (2) (2003) 141–164. doi:10.1145/779359.779362.
- [49] R. Mirzazadeh, A. Ghisi, S. Mariani, Statistical investigation of the mechanical and geometrical properties of polysilicon films through on-chip tests, *Micromachines* 9 (2) (2018). doi:10.3390/mi9020053.

- [50] M. Brossard, A. Barrau, S. Bonnabel, AI-IMU dead-reckoning, *IEEE Transactions on Intelligent Vehicles* 5 (4) (2020) 585–595. doi:10.1109/TIV.2020.2980758.
- [51] F. Saccani, D. Pau, M. Amoretti, In-sensor learning for pressure self-calibration, in: *2024 IEEE Sensors Applications Symposium (SAS)*, 2024, pp. 1–6. doi:10.1109/SAS60918.2024.10636625.
- [52] P. F. Giordano, S. Quqa, M. P. Limongelli, The value of monitoring a structural health monitoring system, *Structural Safety* 100 (2023) 102280. doi:10.1016/j.strusafe.2022.102280.
- [53] S. L. Brunton, B. W. Brunton, J. L. Proctor, E. Kaiser, J. N. Kutz, Chaos as an intermittently forced linear system, *Nature Communications* 8 (1) (2017) 19. doi:10.1038/s41467-017-00030-8.
- [54] L. Rosafalco, P. Conti, EKF-SINDy, <https://github.com/ContiPaolo/SINDy-EKF> (2024).

MOX Technical Reports, last issues

Dipartimento di Matematica
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

- 29/2025** Centofanti, E.; Ziarelli, G.; Parolini, N.; Scacchi, S.; Verani, M. ; Pavarino, L. F.
Learning cardiac activation and repolarization times with operator learning
- 27/2025** Antonietti P.F.; Artoni, A.; Ciaramella, G.; Mazzieri, I.
A review of discontinuous Galerkin time-stepping methods for wave propagation problems
- 28/2025** Ciaramella, G.; Gander, M.J.; Mazzieri, I.
Discontinuous Galerkin time integration for second-order differential problems: formulations, analysis, and analogies
- 24/2025** Bartsch, J.; Borzi, A.; Ciaramella, G.; Reichle, J.
Adjoint-based optimal control of jump-diffusion processes
- 22/2025** Leimer Saglio, C. B.; Pagani, S.; Antonietti P. F.
A p -adaptive polytopal discontinuous Galerkin method for high-order approximation of brain electrophysiology
- 23/2025** Antonietti, P. F.; Caldana, M.; Mazzieri, I.; Re Fraschini, A.
MAGNET: an open-source library for mesh agglomeration by Graph Neural Networks
- 21/2025** Caldera, L., Masci, C., Cappozzo, A., Forlani, M., Antonelli, B., Leoni, O., Ieva, F.
Uncovering mortality patterns and hospital effects in COVID-19 heart failure patients: a novel Multilevel logistic cluster-weighted modeling approach
- 20/2025** Botti, M.; Prada, D.; Scotti, A.; Visinoni, M.
Fully-Mixed Virtual Element Method for the Biot Problem
- 19/2025** Bortolotti, T.; Wang, Y. X. R.; Tong, X.; Menafoglio, A.; Vantini, S.; Sesia, M.
Noise-Adaptive Conformal Classification with Marginal Coverage
- Bortolotti, T.; Wang, Y. X. R.; Tong X.; Menafoglio, A.; Vantini, S.; Sesia, M.
Noise-Adaptive Conformal Classification with Marginal Coverage