This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2020.3042412, IEEE Control Systems Letters

Joint state and dynamics estimation with high-gain observers and Gaussian process models

Mona Buisson-Fenet[1,2,3], Valery Morgenthaler[2], Sebastian Trimpe[3], Florent Di Meglio[1]

***Abstract*— With the rising complexity of dynamical systems generating ever more data, learning dynamics models appears as a promising alternative to physics-based modeling. However, the data available from physical platforms may be noisy and not cover all state variables. Hence, it is necessary to jointly perform state and dynamics estimation. In this paper, we propose interconnecting a high-gain observer and a dynamics learning framework, specifically a Gaussian process state-space model. The observer provides state estimates, which serve as the data for training the dynamics model. The updated model, in turn, is used to improve the observer. Joint convergence of the observer and the dynamics model is proved for high enough gain, up to the measurement and process perturbations. Simultaneous dynamics learning and state estimation are demonstrated on simulations of a mass-spring-mass system.**

***Index Terms*— Machine Learning, Nonlinear systems identification, Observers for nonlinear systems**

I. Introduction

With the recent advances in control engineering, more and more complex dynamical systems are being considered, for which it is often difficult to derive physics-based models. Because of the availability of experimental data, as well as novel algorithms to process it, learning dynamics models directly from that data is an attractive alternative. However, this experimental data often originates from sensor measurements on the physical system, which can be noisy and may not reflect all state variables. On the one hand, reconstituting the full state from noisy, partial measurements falls into the area of state estimation and observer design. For nonlinear systems, this remains a challenging task for which knowledge of the dynamics is usually required [1]. On the other hand, most existing approaches for learning dynamics models require knowing the full state [2]. Therefore, joint state estimation and dynamics learning is needed, a problem often referred to in the machine learning community as inference and learning [3].

The design of observers for nonlinear systems is a complex task for which various approaches have been investigated (see [1] for an overview). We focus on the fairly large class of systems that can be expressed in the so-called observable canonical form (see Sect. II). For these systems, one can design High-Gain Observers (HGOs), which rely on a triangular structure with increasing gain power to compensate for the nonlinearity farther from the measurement. HGOs have been used for a wide variety of applications [4], [5]. In particular, they provide robustness to model uncertainty, as practical convergence can be proved for high enough gain, given only an upper bound on the nonlinearity [6].

Learning dynamics models is also an active research topic. In particular, Gaussian process (GP) state-space models are increasingly used [7], [8]. These nonparametric models exhibit many advantageous properties for learning dynamical systems: they are flexible, data-efficient, probabilistic, and can easily incorporate prior knowledge (see [9] for details). Thanks to their analytical formulation, GPs also allow for theoretical guarantees [10], [11], which is rarely the case for nonparametric machine learning frameworks.

The problem of joint inference and learning for GP state-space models is tackled in its most general form in [3] using variational inference, by modeling the latent states as extra hyperparameters of the GP. The expectation maximization (EM) algortihm is applied: in the first step, measurements are collected and the posterior distribution of the GP is computed. In the second step, all hyperparameters, including the pseudo inputs and outputs representing the evolution of the latent states, are optimized to maximize the data log likelihood. Improvements of this approach have been proposed, e.g., by using more sophisticated loss functions or incorporating additional structure [12], [13]. However, the optimization procedure remains high-dimensional and non-convex. This leads to a high computational burden and a risk of overfitting, which can make the models difficult to train. Furthermore, no theoretical guarantees are yet provided for such methods.

Recent works tackle this problem by combining observer design and data-driven dynamics learning with universal approximators. The model is learned with a neural network using smooth, continuous-time weight update laws [14], [15] or a basis expansion [16], then added to an observer built as a copy of the system plus linear output injection terms. Limited theoretical guarantees have been shown [14], [16], but joint convergence has only been proved if suitable gains can be found by solving a large set of linear matrix inequalities [15]. However, this yields an unusual neural network model for f and an observer with a high number of parameters left to tune, limiting the practical use of the framework.

In this work, we combine the predictive power of machine learning with existing convergence results for state estimation. Our main contribution is the design of a framework for simultaneous state and dynamics estimation, by combining a HGO that estimates the full state from measurements and a

[1]Centre Automatique et Systèmes, Mines ParisTech, PSL University, Paris, France, florent.di_meglio@mines-paristech.fr, mona.buisson@mines-paristech.fr

[2]ANSYS Research Team, ANSYS France, Villeurbanne, France, valery.morgenthaler@ansys.com

[3]Institute for Data Science in Mechanical Engineering, RWTH Aachen University, Aachen, Germany, trimpe@dsme.rwth-aachen.de

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2020.3042412, IEEE Control Systems Letters

GP model that learns the unknown nonlinearity. Convergence guarantees for both the observer and the dynamics model are provided; practical applicability is discussed and demonstrated on simulations. This builds upon the scheme proposed in [17], in which the nonlinearity is considered as a state with partially known dynamics in an extended HGO, and is learned by an identifier satisfying certain requirements. The key difference of our approach is to directly learn a discrete model of the nonlinearity instead of differentiating it and extending the observer. This enables us to deal with controlled systems and input-dependent nonlinearities, and to decrease the error in the data used for regression. Furthermore, it decreases the dimensionality of the observer, which reduces noise amplification by the HGO. We also show that more flexible, non-parametric models such as GPs can learn the input-dependent dynamics while satisfying the smoothness assumptions which are necessary to prove joint convergence.

After formalizing the problem, we present the proposed framework in Sect. II. In Sect. III we show joint convergence of both state and dynamics estimation, then demonstrate our approach on a numerical example in Sect. IV. Comparison to related work and limitations are discussed in Sect. V, before concluding in Sect. VI.

II. Problem formulation and proposed framework

We consider a dynamical system of state $x \in \mathbb{R}^n$, output $y \in \mathbb{R}$, and bounded control input $u \in \mathbb{R}^m$, where $n, m \in \mathbb{N}$. For ease of notation we focus on the single-output case, but all results extend to multiple outputs by concatenation. We assume the following observable canonical form

$$\dot{x} = Ax + Bf(x,u) + D(u) + d, \qquad y = Cx + \epsilon, \tag{1}$$

with f an unknown nonlinearity acting on the n^{th} state x_n, while the rest of the dynamics follows a chain of integrators:

$$A = \begin{pmatrix} 0_{n-1} & I_{n-1} \\ 0 & 0_{n-1}^\intercal \end{pmatrix}, \quad B = \begin{pmatrix} 0_{n-1} \\ 1 \end{pmatrix}, \quad C = \begin{pmatrix} 1 \\ 0_{n-1} \end{pmatrix}^\intercal.$$

The input function $D : \mathbb{R}^m \to \mathbb{R}^n$ is continuous and known while $d \in \mathbb{R}^n$, $\epsilon \in \mathbb{R}$ are unknown disturbances, typically considered deterministic (see Remark 2). All vectors are column vectors; 0_n denotes a vector of n zeros, while I_n is the identity matrix of size n. A broad class of systems can be transformed into this canonical form without knowing f, e.g., all differentially observable systems [1, Sect. 7.1]. Our aim is to compute an estimate $\hat{x}$ of the full state from measurements y, while jointly learning a model $\hat{f}$ of f. We make the following assumptions on (1).

Assumption 1: The true nonlinearity f is Lipschitz continuous of constant L_f. There exist compact sets $\mathcal{X}$ and $\mathcal{U}$ such that $x(t) \in \mathcal{X}$, $u(t) \in \mathcal{U} \ \ \forall\, t \geq 0$. Since f is continuous on a compact space, its norm is also bounded by $f_{\max}$.

The proposed observer follows a cyclic structure, illustrated in Figure 1. During cycle number $j \in \mathbb{N}^*$, the observer produces an estimated state trajectory based on measurements and on the current dynamics model $\hat{f}_{j-1}$. This data is sampled and saved. At the end of the cycle, the model is updated based on the available estimated data. It produces an estimate $\hat{f}_j$, which is then used by the observer for the next cycle.

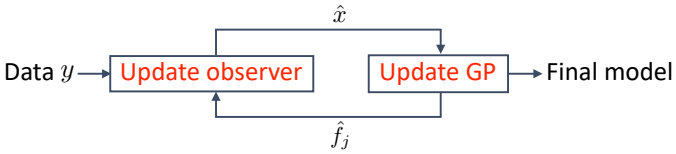


Fig. 1: Structure of the framework: after each cycle, an updated model $\hat{f}_j$ is computed and the observer is adapted.

A. High-gain observer

During cycle number j, the HGO performs state estimation using the current model of the dynamics $\hat{f}_{j-1}$:

$$\dot{\hat{x}} = A\hat{x} + B\hat{f}_{j-1}(\hat{x},u) + D(u) + \Lambda(g)(y - C\hat{x}). \tag{2}$$

The gain is denoted $g > 1$, while $\Lambda(g)$ is the gain matrix following a standard high-gain construction:

$$\Lambda(g) := \begin{pmatrix} gL_1 & g^2L_1 & \cdots & g^nL_n \end{pmatrix}^\intercal, \tag{3}$$

with $L = \begin{pmatrix} L_1 & \cdots & L_n \end{pmatrix}^\intercal \in \mathbb{R}^n$ such that $A - LC$ is Hurwitz. Equation (2) corresponds to the observer block in Figure 1. We make the following assumption, which is ensured by our dynamics model described hereafter.

Assumption 2: For all $j \in \mathbb{N}$, $\hat{f}_j$ is continuous and its norm is bounded by $\hat{f}_{\max}$.

Hence, we can pick $\mathcal{X}$ large enough such that $\hat{x}(t) \in \mathcal{X}$ $\forall\, t \geq 0$. With Assumptions 1–2, the error on the nonlinearity $\hat{f}(x,u) - f(x,u)$ is bounded. Then, as proved in [6] and used in the literature on HGOs, practical convergence in finite time can be shown in the absence of disturbances: for a given error level $\nu > 0$ and a given time $\bar{t} > 0$, there exists a gain g high enough to ensure that for all $t \geq \bar{t}$, $\|\hat{x}(t) - x(t)\| \leq \nu$. Hence, no matter how bad the approximation $\hat{f}$ of f is, as long as an upper bound of the difference is known, practical convergence of the observer can be guaranteed for high enough gain. We leverage this property to build our method.

B. Preliminaries on Gaussian processes

In this paper we focus on Gaussian processes. However, any learning algorithm that satisfies our assumptions can be used. A Gaussian process (GP) is a collection of random variables, any finite subset of which is jointly normally distributed (see [9] for an overview). It is fully characterized by its mean function $m(\cdot)$ and its covariance function $k(\cdot,\cdot)$. Without loss of generality, we assume $m \equiv 0$. Any prior information can be included by substracting it from the output data in order to learn the residuals of this prior. Function properties such as smoothness are encoded in the choice of the kernel $k(\cdot,\cdot)$, which acts as a similarity measure for the values of $f(\cdot)$. At an unobserved point x, given a dataset (X,Y) with Gaussian noise of variance σ_ϵ^2 on the output, the GP prediction of $f(x)$ is normally distributed with posterior mean and variance

$$\mu(x|X,Y) = \underline{k}(x)^\intercal (K + \sigma_\epsilon^2 I)^{-1} Y \tag{4}$$

$$\sigma^2(x|X,Y) = k(x,x) - \underline{k}(x)^\intercal (K + \sigma_\epsilon^2 I)^{-1} \underline{k}(x), \tag{5}$$

where I is the identity matrix, $K = (k(x_i,x_j))_{x_i,x_j \in X}$ is the covariance matrix of X, and $\underline{k}(x) = (k(x_i,x))_{x_i \in X}$. The

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2020.3042412, IEEE Control Systems Letters

kernel k usually depends on some hyperparameters, often obtained by maximizing the data marginal log likelihood.

Assumption 3: The kernel k is differentiable, Lipschitz continuous of constant L_k, and its norm is bounded by $k_{\max}$. This is the case for most commonly used covariance functions, such as the squared exponential. GPs are increasingly used for learning dynamics thanks to their flexibility, data efficiency and analytical formulation. In this work, we use the GP posterior mean as a function approximator to estimate f.

C. Learning method

The state trajectory estimated by the observer is used to learn the dynamics model $\hat{f}$ through batch updates. For each update indexed by $j \in \mathbb{N}^*$, a dataset of length N is constructed by sampling this trajectory with period Δt, starting from the last sample collected at t_j:

$$\begin{aligned} X_j &= \begin{pmatrix} (\hat{x}(t_{j-N}), u(t_{j-N})) & \cdots & (\hat{x}(t_{j-1}), u(t_{j-1})) \end{pmatrix}^\intercal \\ Y_j &= \begin{pmatrix} \hat{x}_n(t_{j-N+1}) & \cdots & \hat{x}_n(t_j) \end{pmatrix}^\intercal . \end{aligned} \tag{6}$$

Since the nonlinearity acts on the n^{th} dimension, x_n is the only output that needs to be collected. The j^{th} update $\hat{f}_j$ is learned from inputs X_j and outputs Y_j, then used in the observer (2) for the next cycle. It can be updated offline since this may necessitate more computing power, and the updates are not necessarily periodic. The model learned from (6) is

$$\mu_j(\cdot|X_j, Y_j) = \underline{k}_j(\cdot)^\intercal (K_j + \sigma_\epsilon^2 I_N)^{-1} Y_j \tag{7}$$

with $K_j = (k(x_i, x_l))_{x_i, x_l \in X_j}$, $\underline{k}_j(x) = (k(x_i, x))_{x_i \in X_j}$. For the formulation of the GP posterior, Gaussian measurement noise of variance σ_ϵ^2 is assumed. However, this need not be the case in reality as we only use the GP as a function approximator and σ_ϵ^2, similarly to other hyperparameters, is chosen in practice for calibration purposes. No assumption on ϵ is needed for the provided theoretical guarantees.

We perform nonlinear regression to estimate the mapping $\mu : (\hat{x}, u) \mapsto \hat{x}_n(t + \Delta t)$. However, $\hat{f}$ in the observer corresponds to the continuous time derivative of $\hat{x}_n$. Hence, we form $\hat{f}_j$ with a Euler differentiation step:

$$\hat{f}_j(\hat{x}, u) = \frac{1}{\Delta t}(\mu_j(\hat{x}, u|X_j, Y_j) - \hat{x}_n). \tag{8}$$

To guarantee the boundedness of $\hat{f}$, we saturate it directly in the observer by imposing $\|\hat{f}\| \leq \hat{f}_{\max}$. The model (8) corresponds to the GP block in Figure 1; given the continuity of (7) and this saturation, it satisfies Assumption 2.

Remark 1: The choice of Δt results from a trade-off: small enough to keep the numerical error from (8) low, large enough to see a real difference between two samples. Learning a discrete model from the estimated trajectory enables us to avoid extending the observer, contrarily to [17].

III. Theoretical guarantees

As stated previously, HGOs are robust to model uncertainty for systems in the observable canonical form. This enables us to decouple the procedures of state estimation and dynamics learning. Indeed, thanks to this robustness, convergence guarantees can still be obtained even in the worst-case scenario, i.e., with maximal but bounded model error, at the cost of a high gain. These convergence properties are then transferred to the dynamics model through its smoothness with respect to the dataset used for learning. Both practical and asymptotic convergence results are provided.

In the following proofs, we focus on the ℓ_2 norm for vectors and matrices, but equivalent bounds can be obtained for any vector norm and its induced matrix norm. We first give a technical result on the smoothness of GP models, showing that the posterior mean is Lipschitz continuous not only with respect to the test point but also to the training dataset.

Lemma 1: Under Assumptions 1–3, the dynamics model $\hat{f}$ as defined in (8) is Lipschitz continuous with respect to each of its variables: $(x, u) \mapsto \hat{f}(x, u|X, Y)$ with constant L_x, and $(X, Y) \mapsto \hat{f}(x, u|X, Y)$ with constant L_z.

Sketch of proof: Assumptions 1–3 yield upper bounds on k and Y. These and the Lipschitz continuity of k directly yield the first claim. The sensitivity of the entries of K with respect to each entry of X is then bounded by elementary algebraic inequalities, and the second claim is obtained similarly. ■

Since the dataset (X_j, Y_j) is constructed from state estimation samples, the error in this data directly depends on the state estimation error. This corresponds to the stability requirement in [17]. Then, Lemma 1 guarantees that any state estimation error in the dataset is smoothly transferred to the obtained model. This is essential to obtain stability guarantees, and corresponds to the regularity requirement in [17]. While we focus on GPs, any learning algorithm satisfying Lemma 1 based on a dataset constructed as in (6) can be used in our framework, to produce an observer and a dynamics model that can both be used for further control tasks.

A. Practical convergence

We denote $d_g = \begin{pmatrix} g^{-1}d_1 & \cdots & g^{-n}d_n \end{pmatrix}$, $\bar{x} = (x, u)$, $\hat{\bar{x}} = (\hat{x}, u)$ and $\|x\|_{\bar{t}} = \max\{\|x(t)\| \mid t \leq \bar{t}\}$. At a given $j \in \mathbb{N}^*$, $\hat{f}_j^*$ is the nonlinearity that would have been learned if the true data X_j^*, Y_j^* had been available, i.e., if the state $x(t)$ had been directly available for sampling according to (6) instead of its estimate $\hat{x}(t)$. The prediction error is written as

$$\varepsilon_j(\cdot) = \hat{f}_j(\cdot) - f(\cdot), \tag{9}$$

while $\varepsilon_j^*(\cdot) = \hat{f}_j^*(\cdot) - f(\cdot)$ is the optimal prediction error that would have been obtained if the true state had been available instead of an estimate. We now state a first result on the practical convergence of the proposed framework.

Theorem 1: For system (1)–(2) with (8) under Assumptions 1–3, for any given error level $\nu > 0$, any time $\bar{t} > 0$, there exists a gain g^* large enough such that for all $g \geq g^*$, $t \geq \bar{t}$ and $j \in \mathbb{N}^*$ such that $\bar{t} \leq t_j \leq t$, in the absence of disturbances ($d \equiv \epsilon \equiv 0$), we have for any fixed $\bar{x} \in \mathcal{X} \times \mathcal{U}$:

$$\max\left\{\|\hat{x}(t) - x(t)\|, \|\hat{f}_j(\bar{x}) - \hat{f}_j^*(\bar{x})\|\right\} \leq \nu. \tag{10}$$

Proof: The proof follows three steps. During a given cycle indexed by some fixed $j \in \mathbb{N}^*$, with $t_j \geq t \geq t_{j-1}$, the model error can be bounded by the maximal error: $\|\hat{f}_{j-1}(\cdot) - f(\cdot)\| \leq \hat{f}_{\max} + f_{\max}$. Using existing proofs of the practical convergence of HGOs under bounded model uncertainty [6], [17], we show

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2020.3042412, IEEE Control Systems Letters

that there exist constants $\rho_1, \rho_2, \rho_3 > 0$ and a gain $g^* > 0$ such that for all $g \geq g^*$, $i \in \{1, ..., n\}$:

$$\|\hat{x}_i(t) - x_i(t)\| \leq \max\big\{g^{i-1}\rho_0 e^{-\rho_1 g t}\|\hat{x}(0) - x(0)\|, \rho_2 g^{i-n-1}, \rho_3 g^{i-1}\|(d_g, \epsilon)\|_t\big\} := B_i(t). \quad (11)$$

The key then lies in bounding the error on X_j and Y_j:

$$\|X_j - X_j^*\| \leq \sum_{l=j-N}^{j-1}\sum_{i=1}^{n}\|\hat{x}_i(t_l) - x_i(t_l)\|, \quad (12)$$

$$\|Y_j - Y_j^*\| \leq \sum_{l=j-N+1}^{j}\|\hat{x}_n(t_l) - x_n(t_l)\|. \quad (13)$$

Therefore, the error on the input and output datasets used to learn $\hat{f}_j$ decreases as the state estimation error decreases, with a delay of N time steps corresponding to the time before earlier samples with a larger error are forgotten. Applying Lemma 1 then (12)-(13) and (11) yields for any $\bar{x} \in \mathcal{X} \times \mathcal{U}$:

$$\|\hat{f}_j(\bar{x}) - \hat{f}_j^*(\bar{x})\| \leq L_z N(n+1)\max\big\{\rho_3 g^{n-1}\|(d_g, \epsilon)\|_{t_j}, \frac{\rho_2}{g}, g^{n-1}\rho_0 e^{-\rho_1 g t_{j-N}}\|\hat{x}(0) - x(0)\|\big\}. \quad (14)$$

Combining (11) and (14) as such leads to a joint practical convergence result, with an additional term corresponding to the disturbances. In the absence of disturbances, i.e., with $d \equiv \epsilon \equiv 0$, this concludes the proof. ■

B. Asymptotic convergence

Theorem 1 shows that the practical convergence guarantees obtained for HGOs with bounded nonlinearity extend to the complete error system. Both the state estimation error and the error made by the dynamics model due to seeing only estimated instead of true data can be made arbitrarily small arbitrarily fast, up to the disturbances, at the cost of a high gain. We now present an asymptotic convergence result arising from the practical convergence of HGOs in the presence of model uncertainty, by bounding this uncertainty depending on the data, the test point, and the optimal ε^*.

Theorem 2: For system (1)–(2) with (8) under Assumptions 1–3, there exist constants $c, c', c'' > 0$ and a gain $g^* > 0$ such that $\forall\ g \geq g^*,\ \forall\ i \in \{1, ..., n\}$ and for any fixed test point $\bar{x} \in \mathcal{X} \times \mathcal{U}$, we have

$$\limsup_{t\to\infty}\|\hat{x}_i(t) - x_i(t)\| \leq c\max\big\{ g^{i-n-1}\limsup_{t+j\to\infty}\|\varepsilon_j^*(x,u)\|, g^{i-1}\limsup_{t\to\infty}\|(d_g,\epsilon)\|\big\}, \quad (15)$$

$$\limsup_{t+j\to\infty}\|\hat{f}_j(\bar{x}) - \hat{f}_j^*(\bar{x})\| \leq \max\big\{ c' g^{-1}\limsup_{t+j\to\infty}\|\varepsilon_j^*(x,u)\|, c'' g^{n-1}\limsup_{t\to\infty}\|(d_g,\epsilon)\|\big\}. \quad (16)$$

Proof: First, using definition (9) and Lemma 1, we show that for any fixed $j \in \mathbb{N}^*$, $(x, \hat{x}, u) \in \mathcal{X} \times \mathcal{X} \times \mathcal{U}$, there exists $c_1 > 0$ such that

$$\begin{aligned}\|\hat{f}_{j-1}(\hat{x}, u) - f(x, u)\| &\leq \|\hat{f}_{j-1}(\hat{x}, u) - \hat{f}_{j-1}^*(x, u)\| + \|\varepsilon_{j-1}^*(x, u)\| \\ &\leq c_1\big(\|\hat{x} - x\| + \|X_{j-1} - X_{j-1}^*\| + \|Y_{j-1} - Y_{j-1}^*\| \\ &\quad + \|\varepsilon_{j-1}^*(x, u)\|\big). \end{aligned} \quad (17)$$

Fig. 2: Schematics of the mass-spring-mass system.

Then, using (17) in the proof of practical convergence of the HGO with g large enough yields a bound similar to (11), but involving $\|X_{j-1} - X_{j-1}^*\| + \|Y_{j-1} - Y_{j-1}^*\| + \|\varepsilon_{j-1}^*(x, u)\|$ instead of the constant term in ρ_2. Going to the limit in both time and number of cycles and using (12)–(13) yields the first claim. Lemma 1 yields the second claim. ■

Theorem 2 bounds the difference between $\hat{f}$ and $\hat{f}^*$ that would have been obtained using true instead of estimated data for learning, without any assumption on the fit of the GP. If ε^* is zero, both the state estimation and the prediction error are input-to-state stable with respect to the disturbances. If there are also no disturbances, they converge asymptotically. Since GPs with universal kernels are universal function approximators, ε^* can get very small (it converges up to the numerical errors due to the Euler differentiation) as the number of samples grows to infinity, if the samples are densely distributed over $\mathcal{X} \times \mathcal{U}$, i.e., if the state-action space is well explored. However, if the hyperparameters of the GP do not enable a good fit or if the data is not rich enough, then ε^* may be large. We further note that it is also possible to bound $\|\hat{f}_j(\bar{x}) - f(\bar{x})\|$ similarly to (16) using the same error decomposition as (17): a term $\varepsilon_j^*(\bar{x})$ appears, representing the optimal error due to the GP model at the test point $\bar{x}$.

Remark 2: Nowhere do we use the form of d and ϵ. In principle, they could also be realizations of stochastic processes. However, Theorem 2 may then not be as meaningful as, depending on the process, $\limsup$ may not exist.

Remark 3: The proposed framework is modular: any regressor $\hat{f}$ can be used seamlessly instead of the GP. If Assumptions 1–2 and Lemma 1 are satisfied, the same theoretical guarantees will hold. This is the case for models of form $\hat{f}(\cdot) = \theta^\intercal\sigma(\cdot)$, where θ is a parameter to be learned from data following a Lipschitz procedure (such as recursive least squares), and σ is a known, Lipschitz feature vector. Other observer designs providing guarantees similar to HGOs could also be considered, e.g., sliding mode observers [18].

Remark 4: In practice, one can learn the residuals model $\mu_{res} : (\hat{x}, u) \mapsto \hat{x}_n(t+\Delta t) - \hat{x}_n - \Delta t D_n(u)$ instead of μ, and use $\mu(\hat{x}, u) = \mu_{res}(\hat{x}, u) + \hat{x}_n + \Delta t D_n(u)$ for prediction. This eases the training process by incorporating prior knowledge into the regression problem, while minimally changing the form of Y_j, which leads to a factor $(n+2)$ instead of $(n+1)$ in (14) and maintains the theoretical guarantees.

IV. Numerical simulation

We demonstrate the performance of the proposed approach on a mass-spring-mass system with a nonlinear spring, as illustrated in Figure 2, and motivated by series-elastic actuator, e.g., [19]. We assume the system can be described by

$$m_1\ddot{x}_1 = f_k(x_2 - x_1), \quad m_2\ddot{x}_2 = -f_k(x_2 - x_1) + u, \quad (18)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2020.3042412, IEEE Control Systems Letters

where x_1, x_2 are the positions of the two objects, $m_1 = m_2 = 1$ are their masses, and $f_k(\cdot)$ is some unknown nonlinear function representing the spring dynamics.

Assuming system (18) is differentially observable of order 4 (see [1] and references), it can be transformed into the observable canonical form (1). This is done by introducing a new state $z \in \mathbb{R}^4$, taking $z_1 = x_1$ and computing the successive derivatives of z_1. As in (1), this yields

$$\begin{aligned} \dot{z} =& Az + Bf(z,u) + d \\ y =& Cz + \epsilon, \end{aligned} \tag{19}$$

where f is an unknown nonlinearity. The proposed approach can directly be applied to (19) without further knowledge about f. In our simulations, we use $f_k(\cdot) = k_1(\cdot) + k_2(\cdot)^3$ with $k_1 = 0.3$, $k_2 = 0.1$ the spring constants. This yields for the true system in observable canonical form:

$$\begin{aligned} f(z,u) =& \frac{3k_2}{m_1 m_2}(u - (m_1+m_2)z_3)v_1^2 + \frac{6k_2}{m_1}v_1 v_2^2 \\ &+ \frac{k_1}{m_1 m_2}(u - (m_1+m_2)z_3), \end{aligned} \tag{20}$$

$$\alpha = \sqrt[3]{\frac{m_1 z_3}{2k_2} + \sqrt{\left(\frac{k_1}{3k_2}\right)^3 + \left(\frac{m_1 z_3}{2k_2}\right)^2}},$$

$$\beta = \sqrt[3]{\frac{m_1 z_3}{2k_2} - \sqrt{\left(\frac{k_1}{3k_2}\right)^3 + \left(\frac{m_1 z_3}{2k_2}\right)^2}},$$

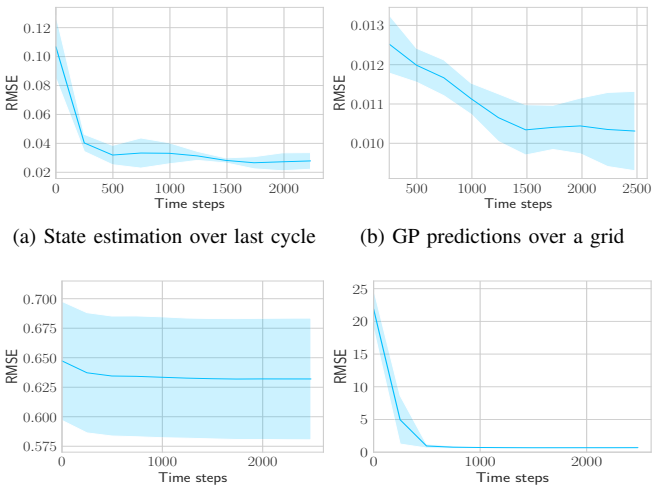
$$v_1 = \alpha + \beta, \qquad v_2 = \frac{z_4}{\frac{k_1}{m_1} + \frac{3k_2}{m_1}v_1^2}.$$

We simulate (19) with d, ϵ Gaussian noise of standard deviation $\sigma_d = \sigma_\epsilon = 10^{-4}$ for ten cycles of 15 seconds each, sampled at $\Delta t = 0.06s$. We set $u(t) = 0.4\cos(1.2t)$, $N = 3000$, $g = 10$, $L = (5,5,3,1)^\intercal$, and $\hat{f}_0 \equiv 0$. We use a squared exponential kernel whose hyperparameters are fixed by maximizing the marginal log likelihood on a subset of data offline. We run 10 simulations from 10 initial conditions with $x_1 \in [0, 0.1]$, $x_2 \in [0.1, 0.2]$, and $\dot{x_1}, \dot{x_2} \in [-0.005, 0.005]$[1]. For each, we start by precomputing a grid of random states and inputs, along with 50 test trajectories of 200 time steps, using a random initial state and one of three control strategies: random control, $u(t) = 0.4\cos(1.2t)$, or $u(t) = 0$.

In each simulation, we evaluate both the observer and the model at the end of every cycle. The observer (2) containing the current model is evaluated by computing the root mean square error (RMSE) between true and estimated trajectory over the last cycle, but also over the test trajectories, given $\hat{z}(t_0) = (y(t_0), 0, 0, 0)^\intercal$ and $y(t)$. The model (8) is evaluated by computing the RMSE of one step ahead predictions over the precomputed grid, and the RMSE of the predicted test trajectories, given the initial state but no measurements.

We observe joint convergence of the observer and the dynamics model. The error in all considered metrics decreases over time, as depicted in Figure 3. The numbers themselves are not necessarily meaningful, but their decreasing behavior and

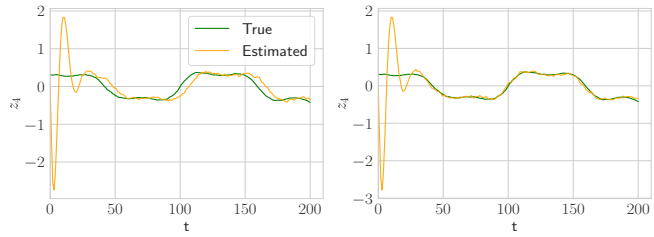
[1]Code available at `https://github.com/monabf/joint_state_dynamics_estimation_HGOs_GPs.git`



(a) State estimation over last cycle (b) GP predictions over a grid

(c) Estimation of test trajectories (d) Prediction of test trajectories

Fig. 3: RMSE of metrics over 10 simulations of (19) (mean $\pm$ 2 standard deviations).



(a) Before learning (b) After ten cycles

Fig. 4: Estimation of a test trajectory of z_4 (random control).

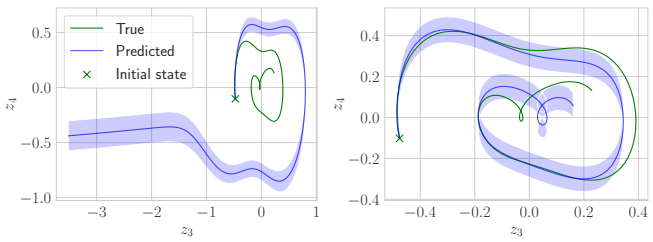
the visual results before and after a few cycles are significant. The variance is due to having different initial conditions, evaluation grids and test trajectories for each run rather than a different performance of the method. The remaining error is caused by the measurement and process noise, along with the irreducible model error given the available data. A test trajectory as estimated by the observer is presented in Figure 4. Before the model is learned (a), the observer's estimates are delayed compared to the true state, because the observer has to wait for correction from the measurements. Once the model has been learned (b), the observer can anticipate and produce accurate estimates without delay. The phase portrait of another test trajectory predicted by the dynamics model is also depicted in Figure 5. It shows the final model can predict the first 100 time steps accurately, then slowly deviates.

Note that the same simulations were also run with the method proposed in [17]. By avoiding to differentiate $\hat{f}$, our method introduces less bias in the data used for regression, and the HGO has a lower dimensionality. As expected, the results seem to indicate that a discrete model learned directly from sampled trajectories yields a better dynamics model, which in turn yields a better observer. This difference grows with the dimensionality and the complexity of the considered nonlinearity.

V. Discussion

In this paper, we propose a framework for joint state and dynamics estimation of dynamical systems in the observable

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2020.3042412, IEEE Control Systems Letters



(a) After one cycle (b) After ten cycles

Fig. 5: Prediction of a test trajectory of z_4 against z_3 with sine control (mean $\pm$ 2 standard deviations).

canonical form. Though the method shows promising results, it has some limitations that require further investigation. On the observer side, Theorems 1 and 2, like all theoretical guarantees provided by HGOs, only hold for high enough gain. However, using a large gain can be prohibitive in practice, mostly in high dimensions or with high measurement noise, as HGOs suffer from peaking and from noise amplification. This can be seen in Figure 4, where peaking is present and the measurement noise is already visible though it was rather low in the simulations ($\sigma_\epsilon = 10^{-4}$), making it difficult to deal with much higher noise. These effects can be mitigated by changing the gain [20] or using multiple HGOs [5], [21]. In future work, these could be combined with our method to allow for higher noise levels. We are also limited by the observable canonical form: many systems can be transformed into this form without knowledge of the dynamics, as in Sect. IV, but the transformation back into the original coordinates remains unknown. In future work, we plan on extending our ideas to more general classes of systems, for example by replacing the HGO with another observer.

On the learning side, we assume $\hat{f}$ is Lipschitz continuous with respect to the training data. This is the case for GPs with fixed hyperparameters. However, as soon as a non-convex optimization procedure is involved, e.g., for online hyperparameter tuning or for training a neural network instead of a GP, this assumption is not satisfied. In this case, Theorems 1 and 2 do not hold. In practice, such tools can often still be used while maintaining performance. Another limitation of GPs is the dimensionality: as the computation of the posterior scales at $O(N^3)$, the GP model can only deal with up to about 10^4 data samples. The sliding window of length N used in (6) deals with this issue but leads to the loss of possibly useful data. Other methods such as sparse GP approximations [22] can be more efficient in practice, but whether the theoretical guarantees extend is open.

VI. Conclusion and Outlook

Due to the imperfect state data provided by most physical platforms, joint estimation of state and dynamics is at the core of dynamics model learning from experimental data. This is a challenging problem in general: few approaches exist, even fewer provide convergence guarantees. In this paper, we propose a framework for joint state estimation and dynamics learning of nonlinear systems in the observable canonical form. A high-gain observer estimates the state trajectory, which is used for learning the nonlinearity with a non-parametric Gaussian process model. Practical and asymptotic convergence of both the state and dynamics estimation can be guaranteed, so that after convergence, the observer and the dynamics model can be used for further control tasks. Simultaneous model learning and improved state estimation are demonstrated on a numerical example. In future work, we plan on extending our approach to further ease application.

References

[1] P. Bernard, "Observer Design for Nonlinear Systems," in *Lecture Notes in Control and Information Sciences*, vol. 479, Springer International Publishing, 2019.
[2] D. Nguyen-Tuong and J. Peters, "Model learning for robot control: A survey," *Cognitive Processing*, vol. 12, pp. 319–340, 2011.
[3] R. Turner, M. P. Deisenroth, and C. E. Rasmussen, "State-Space Inference and Learning with Gaussian Processes," *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, pp. 868–875, 2010.
[4] A. M. Dabroom and H. K. Khalil, "Discrete-time implementation of high-gain observers for numerical differentiation," *International Journal of Control*, vol. 72, no. 17, pp. 1523–1537, 1999.
[5] D. Astolfi, L. Marconi, L. Praly, and A. R. Teel, "Low-power peaking-free high-gain observers," *Automatica*, vol. 98, pp. 169–179, 2018.
[6] A. Tornambé, "High-gain observers for non-linear systems," *International Journal of Systems Science*, vol. 23, no. 9, pp. 1475–1489, 1992.
[7] J. Kocijan, *Modelling and Control of Dynamic Systems Using Gaussian Process Models*. Springer, 2016.
[8] M. Buisson-Fenet, F. Solowjow, and S. Trimpe, "Actively Learning Gaussian Process Dynamics," in *Proceedings of the 2nd Conference on Learning for Dynamics and Control*, vol. 120, pp. 5–15, 2020.
[9] C. E. Rasmussen and C. K. Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2006.
[10] J. Umlauft and S. Hirche, "Learning stochastically stable Gaussian process state–space models," *IFAC Journal of Systems and Control*, vol. 12, 2020.
[11] A. Lederer, J. Umlauft, and S. Hirche, "Uniform Error Bounds for Gaussian Process Regression with Application to Safe Control," in *Advances in Neural Information Processing Systems*, pp. 657–667, 2019.
[12] A. Doerr, C. Daniel, M. Schiegg, D. Nguyen-Tuong, S. Schaal, M. Toussaint, and S. Trimpe, "Probabilistic recurrent state-space models," *Proceedings of the 35th International Conference on Machine Learning*, pp. 1280–1289, 2018.
[13] S. Melchior, F. Berkenkamp, S. Curi, and A. Krause, "Structured Variational Inference in Unstable Gaussian Process State Space Models," in *Proceedings of the 2nd Conference on Learning for Dynamics and Control*, vol. 120, pp. 147–157, 2020.
[14] F. Abdollahi, H. A. Talebi, and R. V. Patel, "A stable neural network observer with application to flexible-joint manipulators," *IEEE Transactions on Neural Networks*, vol. 17, no. 1, pp. 118–129, 2006.
[15] A. Chakrabarty, A. Zemouche, R. Rajamani, and M. Benosman, "Robust Data-Driven Neuro-Adaptive Observers with Lipschitz Activation Functions," in *Proceedings of the 58th IEEE Conference on Decision and Control*, pp. 2862–2867, 2019.
[16] A. Chakrabarty and M. Benosman, "Safe Learning-based Observers for Unknown Nonlinear Systems using Bayesian Optimization," *arXiv preprint arXiv:2005.05888*, pp. 1–20, 2020.
[17] M. Bin and L. Marconi, "Model identification and adaptive state observation for a class of nonlinear systems," *IEEE Transactions on Automatic Control*, 2020.
[18] Y. Shtessel, C. Edwards, L. Fridman, and A. Levant, "Sliding Mode Control and Observation," in *Control Engineering*, Birkhäuser, New York, 2016.
[19] I. Thorson and D. Caldwell, "A nonlinear series elastic actuator for highly dynamic motions," *IEEE International Conference on Intelligent Robots and Systems*, pp. 390–394, 2011.
[20] J. H. Ahrens and H. K. Khalil, "High-Gain Observers in the Presence of Measurement Noise: A Switched-Gain Approach," in *Proceedings of the 17th IFAC World Congress*, vol. 41, pp. 7606–7611, 2008.
[21] K. Esfandiari and M. Shakarami, "Bank of High-Gain Observers in Output Feedback Control: Robustness Analysis Against Measurement Noise," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–12, 2019.
[22] J. Quiñonero-Candela and C. E. Rasmussen, "A unifying view of sparse approximate Gaussian process regression," *Journal of Machine Learning Research*, vol. 6, pp. 1939–1959, 2005.