Postprint. To appear in the *International Journal of Control*.
DOI: https://doi.org/10.1080/00207179.2018.1557348

Finite-horizon integration for continuous-time identification: bias analysis and application to variable stiffness actuators

Dario Piga[a*]

[a]*Dalle Molle Institute for Artificial Intelligence, Via Cantonale 2C, 6928 Manno, Switzerland*

(*Received 15 Jun 2018; accepted 30 Nov 2018*)

Direct identification of continuous-time dynamical models from sampled data is now a mature discipline, which is known to have many advantages with respect to indirect approaches based on the identification of discretized models. This paper faces the problem of continuous-time identification of linear time-invariant systems through finite-horizon numerical integration and least-square estimation. The bias in the least-squares estimator due to the noise corrupting the signal observations is quantified, and the benefits of numerical integration in the attenuation of this bias are discussed. An extension of the approach which combines numerical integration, least-squares estimation and particle swarm optimization is proposed for the identification of nonlinear systems and nonlinear-in-the-parameter models, and then applied to the estimation of the torque-displacement characteristic of a commercial variable stiffness actuator driving a one-degree-of-freedom pendulum.

Keywords: Continuous-time identification; least-squares methods; parameter estimation; variable stiffness actuators.

1. Introduction

The art and the science of building mathematical models of dynamical systems from experimental data, commonly referred to as *system identification*, plays an essential role when deriving models from first-principle laws of physics is too complex and laborious, or when some physical parameters are unknown and they need to be estimated from experiments. Although many physical systems have a natural representation in a *continuous-time* (CT) domain, most of the methodologies for system identification proposed in the literature, such as prediction error methods and maximum-likelihood estimation (Ljung, 1999), have been developed for *discrete-time* (DT) models and, although identification of continuous-time models is now a well established research area, identification of DT models is still predominant in the systems and control community. On the one hand, DT models provide a simple description, in terms of difference equations, of the relationship among the sampled observations of the system's signals, and identification methods for discrete-time models do not need to reconstruct continuous-time derivatives of the signals from noisy samples. On the other hand, there are many advantages in a *direct* identification of continuous-time models, as discussed in (Garnier & Young, 2014a, 2012). First, continuous-time models can provide physical insights about the system under consideration, as the model's parameters have often a physical meaning (*e.g.*, mass, reaction times, friction coefficients, etc.). Second, direct CT identification methods can efficiently handled non-uniformly data, whereas DT models are associated to a fixed sampling time. Third, discrete-time identification methodologies perform poorly in the case of high sampling frequency, as the poles of the DT model lie very close to the unit circle in the complex plane, leading to numerically ill-conditioned problems both in the parameters' estimate and in

*Corresponding author. Email: dario.piga@supsi.ch

the final conversion of the estimated DT model to a CT one. Exhaustive reviews and successful applications of direct continuous-time identification methods can be found in (Garnier, 2015; Garnier, Gilson, Young, & Huselstein, 2007; Garnier, Mensler, & Richard, 2003; Mercère, Palsson, & Poinot, 2011; Rao & Unbehauen, 2006; P. C. Young & Garnier, 2006), in the special issue (Garnier & Young, 2014b), and in the book (Garnier & Wang, 2008). Continuous-time identification algorithms are also implemented in widely-used Matlab toolboxes, such as the *System Identification Toolbox* (Lennart & Singh, 2012) and the *CONTSID Toolbox* (Padilla, Garnier, & Gilson, 2015).

From a general perspective, the direct CT identification methods available in the literature can be classified into three main groups: *state-variable filters* methods (Bastogne, Garnier, & Sibille, 2001; P. Young & Jakeman, 1980), which prefilter the input and output signals through CT linear filters with a predefined bandwidth; *numerical integration* methods (Jemni & Trigeassou, 1996; Sagara & Zhao, 1990; Sinha & Rao, 1991), where the *ordinary differential equation* (ODE) that describes the system's dynamics is integrated, thus avoiding the computation of the time derivatives of the signals; and methods based on *modulating functions* (Daniel-Berhe & Unbehauen, 1998; Lataire, Pintelon, Piga, & Tóth, 2017; Patra & Unbehauen, 1995; Shanghong & Jue, 2004), where the ODE is pre-multiplied by the so-called modulating functions (like Fourier and Hartley functions), and then integrated. The advantages of using modulating functions is that the integrals of the modulated signals can be computed just based on the known derivatives of the modulating function.

This paper presents a numerical-integration based approach for CT identification of *linear time-invariant* (LTI) systems, where the coefficients of the n-order ODE describing the system's dynamics are computed through a *least-squares* (LS) algorithm, after integration of the ODE n times over a finite-time window of length T. The proposed identification approach resembles the linear-integral-filter method (Sagara & Zhao, 1990), in the sense that input and output signals are integrated over a moving window, with no need to estimate the unknown initial conditions. Unlike the approach in (Sagara & Zhao, 1990), which uses *instrumental variables* (IV) to guarantee consistency of the the estimated model parameters, simple least-squares are used in our approach, with no need to compute the *instruments*[1]. The (asymptotic) bias in the least-squares estimator is quantified, and it is shown how finite-horizon numerical integration can significantly reduce this bias w.r.t. to the bias affecting the estimate of the parameters of the discretized model. An intuitive explanation of the role played by the integration horizon T in the reduction of the bias is provided in Section 4 using a simple numerical example, and the advantages w.r.t. the identification of the discretized model are highlighted. Although the theoretical bias analysis is valid under the assumption that data are generated by a linear time-invariant systems with an *output-error* model structure, a practical extension of the approach to nonlinear system identification is also presented in Section 5, where a combined use of linear regression and *particle swarm optimization* (PSO) is employed to estimate the torque-displacement characteristic of a commercial *variable stiffness actuator* (VSA) driving a one-degree-of-freedom pendulum.

Overall, the objective of this paper is to show that, although consistency is not guaranteed by least-squares, satisfactory results can be achieved in practice, provided that numerical integration of input and output signals is first performed.

1.1 *Notation*

The following notation will be used throughout the paper: $\mathbb{R}$ and $\mathbb{Z}$ denote the set of real and integer numbers, respectively; $\mathbb{R}^n$ is the space of real vectors of dimension n; $\mathbb{R}^{m,n}$ is the space of real matrices with m rows and n columns. The transpose of a vector $\boldsymbol{\theta} \in \mathbb{R}^n$ is represented as $\boldsymbol{\theta}^T$;

[1]An improper choice of the instruments might lead to high-variance estimates of the model parameters (Söderström & Stoica, 1983).

$\|\boldsymbol{\theta}\|$ is the norm of the vector $\boldsymbol{\theta}$; and $\|\mathbf{A}\|$ is the norm of a matrix $\mathbf{A} \in \mathbb{R}^{m,n}$, defined as

$$\|\mathbf{A}\| = \sup_{\boldsymbol{\theta}\in\mathbb{R}^n, \boldsymbol{\theta}\neq 0} \frac{\|\mathbf{A}\boldsymbol{\theta}\|}{\|\boldsymbol{\theta}\|}.$$

2. Identification of LTI models: problem setting

Consider a continuous-time linear time-invariant system $\mathcal{S}$ described by the ordinary differential equation of order n:

$$x^{(n)}(t) + a^{\mathrm{o}}_{n-1}x^{(n-1)}(t) + \ldots + a^{\mathrm{o}}_0 x(t) = b^{\mathrm{o}}_{n-1}u^{(n-1)}(t) + \ldots + b^{\mathrm{o}}_0 u(t), \tag{1}$$

where $u(t) \in \mathbb{R}$ and $x(t) \in \mathbb{R}$ are the input and the noise-free output of the system at time $t \in \mathbb{R}$, respectively, $x^{(n)}(t)$ is the n-th time derivative of the signal $x(t)$, and $a^{\mathrm{o}}_i, b^{\mathrm{o}}_i \in \mathbb{R}$ (with $i = 0, \ldots, n-1$) are (unknown) parameters characterizing the system $\mathcal{S}$. In order to compact the notation, the parameters a^{o}_i and b^{o}_i are stacked in the $2n$-dimensional vector

$$\boldsymbol{\theta}^{\mathrm{o}} = \begin{bmatrix} a^{\mathrm{o}}_{n-1} & \cdots & a^{\mathrm{o}}_0 & b^{\mathrm{o}}_{n-1} & \cdots & b^{\mathrm{o}}_0 \end{bmatrix}^T.$$

The input $u(t)$ and output $x(t)$ signals are sampled at regular time intervals[2] of length $T_s > 0$, and the observation of $x(t)$ at time kT_s, with $k \in \mathbb{Z}$, is assumed to be corrupted by an additive zero-mean white noise $e(kT_s)$ statistically independent of $x(kT_s)$ and $u(kT_s)$, *i.e.*,

$$\begin{aligned} y(kT_s) =& x(kT_s) + e(kT_s) \quad \forall\, k \in \mathbb{Z}, \\ \mathbb{E}[e(kT_s)] =& 0 \quad \forall\, k \in \mathbb{Z}, \\ \mathbb{E}[e(kT_s)e(k'T_s)] =& \begin{cases} \sigma_e^2 & \text{if } k = k' \\ 0 & \text{otherwise} \end{cases} \quad \forall\, k, k' \in \mathbb{Z}, \\ \mathbb{E}[e(k)x(k')] =& \mathbb{E}[e(k)u(k')] = 0 \quad \forall\, k, k' \in \mathbb{Z}, \end{aligned}$$

where $y(kT_s)$ denotes the noise-corrupted observation of $x(kT_s)$ and σ_e^2 is the variance of $e(kT_s)$.

The goal of the identification problem addressed in the paper is to estimate parameters $\boldsymbol{\theta}^{\mathrm{o}}$ of the continuous-time system $\mathcal{S}$ in (1) based on a sampled data sequence $\{u(kT_s), y(kT_s)\}_{k=0}^{N-1}$, where N denotes the number of observations. A numerical-integration-based identification scheme is described in the following section, an upper bound on the bias in the final parameters' estimate due to the measurement noise is computed, and an illustrative example is used to provide intuitions on the advantages of numerical integration w.r.t. to an indirect identification approach based on the discretization of the differential equation (1).

3. Numerical-integration and least-squares estimate for CT identification

3.1 *Numerical integration of the noise-free ODE*

Instead of considering a discrete-time approximation of the system's dynamics in (1), let us integrate both sides of the ODE (1) n times over a finite-time window of length T, where T is a multiple of the sampling interval T_s, *i.e.*, $T = N_p T_s$, with $N_p \in \mathbb{N}$. In the following, we refer to T (or

[2]The signals $u(t)$ and $x(t)$ are assumed to be sampled at a regular interval just to keep the notation simple. Nevertheless, the identification algorithm discussed in the paper can be also applied to the case of non-uniformly sampled data.

equivalently N_p) as *integration horizon*. Integration of (1) leads to

$$\begin{aligned}
&\int_{t_n}^{t_n+T} \cdots \left[\int_{t_2}^{t_2+T} \left[\int_{t_1}^{t_1+T} x^{(n)}(t)dt\right] dt_1\right] \cdots dt_{n-1} = \\
&= -\, a_{n-1}^{\mathrm{o}} \int_{t_n}^{t_n+T} \cdots \left[\int_{t_2}^{t_2+T} \left[\int_{t_1}^{t_1+T} x^{(n-1)}(t)dt\right] dt_1\right] \cdots dt_{n-1} + \\
&- \ldots - a_0^{\mathrm{o}} \int_{t_n}^{t_n+T} \cdots \left[\int_{t_2}^{t_2+T} \left[\int_{t_1}^{t_1+T} x(t)dt\right] dt_1\right] \cdots dt_{n-1} + \\
&+ b_{n-1}^{\mathrm{o}} \int_{t_n}^{t_n+T} \cdots \left[\int_{t_2}^{t_2+T} \left[\int_{t_1}^{t_1+T} u^{(n-1)}(t)dt\right] dt_1\right] \cdots dt_{n-1} + \\
&+ \ldots + b_0^{\mathrm{o}} \int_{t_n}^{t_n+T} \cdots \left[\int_{t_2}^{t_2+T} \left[\int_{t_1}^{t_1+T} u(t)dt\right] dt_1\right] \cdots dt_{n-1}.
\end{aligned} \tag{2}$$

Since for any integer $n > 0$ the integral $\int_{t_n}^{t_n+T} \cdots \int_{t_2}^{t_2+T} \int_{t_1}^{t_1+T} x^{(n)}(t)dtdt_1 \cdots dt_{n-1}$ is equal to $\sum_{i=0}^{n} (-1)^i \begin{pmatrix} n \\ i \end{pmatrix} x(t_n + (n-i)T)$, eq. (2) becomes

$$\begin{aligned}
&x(t_n + nT) = -\sum_{i=1}^{n} (-1)^i \begin{pmatrix} n \\ i \end{pmatrix} x(t_n + (n-i)T) + \\
&- a_{n-1}^{\mathrm{o}} \int_{t_n}^{t_n+T} \sum_{i=0}^{n-1} (-1)^i \begin{pmatrix} n-1 \\ i \end{pmatrix} x(t_{n-1} + (n-1-i)T)dt_{n-1} + \\
&- a_{n-2}^{\mathrm{o}} \int_{t_n}^{t_n+T} \int_{t_{n-1}}^{t_{n-1}+T} \sum_{i=0}^{n-2} (-1)^i \begin{pmatrix} n-2 \\ i \end{pmatrix} x(t_{n-2} + (n-2-i)T)dt_{n-2}dt_{n-1} + \\
&- \ldots - a_0^{\mathrm{o}} \int_{t_n}^{t_n+T} \ldots \int_{t_2}^{t_2+T} \int_{t_1}^{t_1+T} x(t)dtdt_1 \ldots dt_{n-1} + \\
&+ b_{n-1}^{\mathrm{o}} \int_{t_n}^{t_n+T} \sum_{i=0}^{n-1} (-1)^i \begin{pmatrix} n-1 \\ i \end{pmatrix} u(t_{n-1} + (n-1-i)T)dt_{n-1} + \\
&+ b_{n-2}^{\mathrm{o}} \int_{t_n}^{t_n+T} \int_{t_{n-1}}^{t_{n-1}+T} \sum_{i=0}^{n-2} (-1)^i \begin{pmatrix} n-2 \\ i \end{pmatrix} u(t_{n-2} + (n-2-i)T)dt_{n-2}dt_{n-1} + \\
&+ \ldots + b_0^{\mathrm{o}} \int_{t_n}^{t_n+T} \ldots \int_{t_2}^{t_2+T} \int_{t_1}^{t_1+T} u(t)dtdt_1 \ldots dt_{n-1}.
\end{aligned} \tag{3}$$

3.2 *Output predictor*

Let us now introduce the following parametrized predictor $\tilde{y}(t_n + nT; \boldsymbol{\theta})$ for the output $x(t_n + nT)$, which shares the same structure of $x(t_n + nT)$ in (3) and it depends on past input $u(t)$ and (noisy) output $y(t)$, for t in the open interval $(t_n \;\; t_n + nT)$:

$$
\begin{aligned}
\tilde{y}(t_n + nT; \boldsymbol{\theta}) = &- \sum_{i=1}^{n} (-1)^i \begin{pmatrix} n \\ i \end{pmatrix} y(t_n + (n-i)T) + \\
&- a_{n-1} \int_{t_n}^{t_n+T} \sum_{i=0}^{n-1} (-1)^i \begin{pmatrix} n-1 \\ i \end{pmatrix} y(t_{n-1} + (n-1-i)T) dt_{n-1} + \\
&- a_{n-2} \int_{t_n}^{t_n+T} \int_{t_{n-1}}^{t_{n-1}+T} \sum_{i=0}^{n-2} (-1)^i \binom{n-2}{i} y(t_{n-2} + (n-2-i)T) dt_{n-2} dt_{n-1} + \\
&- \ldots - a_0 \int_{t_n}^{t_n+T} \ldots \int_{t_2}^{t_2+T} \int_{t_1}^{t_1+T} y(t) dt dt_1 \ldots dt_{n-1} + \\
&+ b_{n-1} \int_{t_n}^{t_n+T} \sum_{i=0}^{n-1} (-1)^i \begin{pmatrix} n-1 \\ i \end{pmatrix} u(t_{n-1} + (n-1-i)T) dt_{n-1} + \\
&+ b_{n-2} \int_{t_n}^{t_n+T} \int_{t_{n-1}}^{t_{n-1}+T} \sum_{i=0}^{n-2} (-1)^i \binom{n-2}{i} u(t_{n-2} + (n-2-i)T) dt_{n-2} dt_{n-1} + \\
&+ \ldots + b_0 \int_{t_n}^{t_n+T} \ldots \int_{t_2}^{t_2+T} \int_{t_1}^{t_1+T} u(t) dt dt_1 \ldots dt_{n-1},
\end{aligned} \tag{4}
$$

where $\boldsymbol{\theta} \in \mathbb{R}^{2n}$ is the vector which stacks the parameters a_i and b_i (with $i = 0, \ldots, n-1$) in its rows, *i.e.*,

$$
\boldsymbol{\theta} = [a_{n-1} \quad \cdots \quad a_0 \quad b_{n-1} \quad \cdots \quad b_0]^T .
$$

Using the available sampled data and the forward-Euler method to approximate the integrals $\int_t^{t+T} u(t)dt$ and $\int_t^{t+T} y(t)dt$, *i.e.*,

$$
\begin{aligned}
\int_t^{t+T} u(t)dt &= \int_t^{t+N_pT_s} u(t)dt \approx T_s \sum_{k=0}^{N_p-1} u(t + kT_s), \\
\int_t^{t+T} y(t)dt &= \int_t^{t+N_pT_s} y(t)dt \approx T_s \sum_{k=0}^{N_p-1} y(t + kT_s),
\end{aligned}
$$

the predictor $\tilde{y}(t_n + nT; \boldsymbol{\theta})$ (4) is approximated by the linear regression form:

$$\begin{aligned}
\hat{y}(kT_s + nT; \boldsymbol{\theta}) = &- \sum_{i=1}^{n} (-1)^i \binom{n}{i} y(kT_s + (n-i)T) + \\
&- a_{n-1} T_s \sum_{k_n=0}^{N_p-1} \sum_{i=0}^{n-1} (-1)^i \binom{n-1}{i} y(kT_s + k_n T_s + (n-1-i)T) + \\
&- a_{n-2} T_s^2 \sum_{k_n=0}^{N_p-1} \sum_{k_{n-1}=0}^{N_p-1} \sum_{i=0}^{n-2} (-1)^i \binom{n-2}{i} y(kT_s + k_n T_s + k_{n-1} T_s + (n-2-i)T) + \\
&- \ldots - a_0 T_s^n \sum_{k_n=0}^{N_p-1} \sum_{k_{n-1}=0}^{N_p-1} \cdots \sum_{k_1=0}^{N_p-1} y(kT_s + k_n T_s + k_{n-1} T_s + \cdots k_1 T_s) + \\
&+ b_{n-1} T_s \sum_{k_n=0}^{N_p-1} \sum_{i=0}^{n-1} (-1)^i \binom{n-1}{i} u(kT_s + k_n T_s + (n-1-i)T) + \\
&+ b_{n-2} T_s^2 \sum_{k_n=0}^{N_p-1} \sum_{k_{n-1}=0}^{N_p-1} \sum_{i=0}^{n-2} (-1)^i \binom{n-2}{i} u(kT_s + k_n T_s + k_{n-1} T_s + (n-2-i)T) + \\
&+ \ldots + b_0 T_s^n \sum_{k_n=0}^{N_p-1} \sum_{k_{n-1}=0}^{N_p-1} \cdots \sum_{k_1=0}^{N_p-1} u(kT_s + k_n T_s + k_{n-1} T_s + \cdots k_1 T_s).
\end{aligned} \tag{6}$$

By introducing the regressor

$$\boldsymbol{\phi}(kT_s + nT) = \begin{bmatrix}
-T_s \sum_{k_n=0}^{N_p-1} \sum_{i=0}^{n-1} (-1)^i \binom{n-1}{i} y(kT_s + k_n T_s + (n-1-i)T) \\
-T_s^2 \sum_{k_n=0}^{N_p-1} \sum_{k_{n-1}=0}^{N_p-1} \sum_{i=0}^{n-2} (-1)^i \binom{n-2}{i} y(kT_s + k_n T_s + k_{n-1} T_s + (n-2-i)T) \\
\vdots \\
-T_s^n \sum_{k_n=0}^{N_p-1} \sum_{k_{n-1}=0}^{N_p-1} \cdots \sum_{k_1=0}^{N_p-1} y(kT_s + k_n T_s + k_{n-1} T_s + \cdots k_1 T_s) \\
T_s \sum_{k_n=0}^{N_p-1} \sum_{i=0}^{n-1} (-1)^i \binom{n-1}{i} u(kT_s + k_n T_s + (n-1-i)T) \\
\vdots \\
T_s^n \sum_{k_n=0}^{N_p-1} \sum_{k_{n-1}=0}^{N_p-1} \cdots \sum_{k_1=0}^{N_p-1} u(kT_s + k_n T_s + k_{n-1} T_s + \cdots k_1 T_s)
\end{bmatrix}, \tag{7}$$

the predictor $\hat{y}(kT_s + nT; \boldsymbol{\theta})$ in (6) is written in the compact form

$$\hat{y}(kT_s + nT; \boldsymbol{\theta}) = - \sum_{i=1}^{n} (-1)^i \binom{n}{i} y(kT_s + (n-1)T) + \boldsymbol{\theta}^T \boldsymbol{\phi}(kT_s + nT). \tag{8}$$

Remark 1: For an integration horizon $T = T_s$ (or equivalently, $N_p = 1$), the predictor $\hat{y}(kT_s +$

$nT;\boldsymbol{\theta})$ in (6) reads as:

$$\begin{aligned}
\hat{y}(kT_s+nT_s;\boldsymbol{\theta}) = &-\sum_{i=1}^{n}(-1)^i\begin{pmatrix} n \\ i \end{pmatrix} y(kT_s+(n-i)T_s)+ \\
&- a_{n-1}T_s\sum_{i=0}^{n-1}(-1)^i\begin{pmatrix} n-1 \\ i \end{pmatrix} y(kT_s+(n-1-i)T_s)+ \\
&- a_{n-2}T_s^2\sum_{i=0}^{n-2}(-1)^i\begin{pmatrix} n-2 \\ i \end{pmatrix} y(kT_s+(n-2-i)T_s)+ \\
&- \ldots - a_0T_s^n y(kT_s)+ \\
&+ b_{n-1}\sum_{i=0}^{n-1}(-1)^i\begin{pmatrix} n-1 \\ i \end{pmatrix} u(kT_s+(n-1-i)T_s)+ \\
&+ b_{n-2}T_s^2\sum_{i=0}^{n-2}(-1)^i\begin{pmatrix} n-2 \\ i \end{pmatrix} u(kT_s+(n-2-i)T_s)+ \\
&+ \ldots + b_0T_s^n u(kT_s). \qquad (9)
\end{aligned}$$

The predictor (9) represents the approximation of the noisy output $y(kT_s+nT)$ obtained by first discretizing the ODE (1) using a forward-Euler method with a sampling time T_s, and finally replacing the noise-free output $x(kT_s)$ with the noisy observation $y(kT_s)$. In other words, eq. (9) represents a discrete-time approximation of the ODE (1) based on sampled inputs and (noisy) outputs. ■

3.3 *Least-squares estimate*

A least-squares estimate of the model parameters $\boldsymbol{\theta}$ can now be computed based on the linear regression form (6), by minimizing the norm of the prediction error

$$\varepsilon(kT_s+nT,\boldsymbol{\theta}) = y(kT_s+nT) - \hat{y}(kT_s+nT;\boldsymbol{\theta}). \qquad (10)$$

Specifically, the least-square estimator $\hat{\boldsymbol{\theta}}_{\text{LS}}$ is given by

$$\hat{\boldsymbol{\theta}}_{\text{LS}}(N_p) = \min_{\boldsymbol{\theta}}\sum_{k=0}^{\tilde{N}}\left(y(kT_s+nT) - \hat{y}(kT_s+nT;\boldsymbol{\theta})\right)^2, \qquad (11)$$

with $\tilde{N} = N - nN_p - 1$. Note that the dependence of $\hat{\boldsymbol{\theta}}_{\text{LS}}$ on the integration horizon N_p is made explicit in (11). By stacking the "filtered" output y_f, defined as

$$y_f(kT_s+nT) = \sum_{i=0}^{n}(-1)^i\begin{pmatrix} n \\ i \end{pmatrix} y(kT_s+(n-i)T), \qquad (12)$$

and the regressor $\boldsymbol{\phi}(kT_s+nT)$, with $k=0,\ldots,\tilde{N}$, in the rows of a vector $\boldsymbol{Y}_f \in \mathbb{R}^{\tilde{N}}$ and of a matrix

$\mathbf{\Phi} \in \mathbb{R}^{\tilde{N},2n}$, respectively, *i.e.*,

$$\boldsymbol{Y}_f = \begin{bmatrix} y_f(nT) \\ y_f(T_s + nT) \\ \vdots \\ y_f((N-1)T) \end{bmatrix}, \quad \mathbf{\Phi} = \begin{bmatrix} \boldsymbol{\phi}(nT)^T \\ \boldsymbol{\phi}(T_s + nT)^T \\ \vdots \\ \boldsymbol{\phi}((N-1)T_s)^T \end{bmatrix}, \tag{13}$$

the analytic expression of the least-square estimator $\hat{\boldsymbol{\theta}}_{\text{LS}}$ can be written in the compact form

$$\hat{\boldsymbol{\theta}}_{\text{LS}}(N_p) = \min_{\boldsymbol{\theta}} \|\boldsymbol{Y}_f - \mathbf{\Phi}\boldsymbol{\theta}\|^2 = \left(\mathbf{\Phi}^T\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^T\boldsymbol{Y}_f. \tag{14}$$

The least-squares estimate $\hat{\boldsymbol{\theta}}_{\text{LS}}$ is asymptotically biased because of the correlation between the regressors $\boldsymbol{\phi}(kT_s)$ and the random noise corrupting the filtered output $y_f(kT_s)$. In the following section, we quantify the asymptotic bias and we show how the integration horizon N_p plays an important role in the reduction of this bias.

Remark 2: A consistent estimate of the system's parameters $\boldsymbol{\theta}^{\text{o}}$ can be achieved using *instrumental-variables* (IV) methods (Söderström & Stoica, 1983). A straightforward application of IV methods to the considered problem leads to the following modification of the the least-squares estimator:

$$\hat{\boldsymbol{\theta}}_{\text{IV}} = \left(\mathbf{Z}^T\mathbf{\Phi}\right)^{-1}\mathbf{Z}^T\boldsymbol{Y}_f, \tag{15}$$

where $\mathbf{Z} \in \mathbb{R}^{\tilde{N},2n}$ is a matrix containing the so called *instruments*. The matrix $\mathbf{Z}$ is chosen by the user with the main requirement that the instruments are not correlated with the output noise $e(kT_s)$. This guarantees consistency of the IV estimator $\hat{\boldsymbol{\theta}}_{\text{IV}}$. However, the variance of the estimate $\hat{\boldsymbol{\theta}}_{\text{IV}}$ strongly depends on the chosen instruments, and iterative algorithms to refine the instruments should be used to reduce the variance of the estimated parameters $\hat{\boldsymbol{\theta}}_{\text{IV}}$ (Söderström & Stoica, 1983; P. Young & Jakeman, 1980). The use of IV methods is not addressed in this paper, which instead just aims to show how the bias in the least-squares estimate can be reduced thanks to numerical integration of the input and of output signals. ■

3.4 *Asymptotic bias in the least-squares estimate*

In order to analyses the asymptotic behaviour of the least-squares estimator $\hat{\boldsymbol{\theta}}_{\text{LS}}$ as the number N of observations goes to infinity, let us first factorize the noisy filtered output $y_f(kT_s + nT)$ in (12) as follows:

$$y_f(kT_s + nT) = \underbrace{\sum_{i=0}^{n}(-1)^i \begin{pmatrix} n \\ i \end{pmatrix} x(kT_s + (n-i)T)}_{x_f(kT_s+nT)} + \underbrace{\sum_{i=0}^{n}(-1)^i \begin{pmatrix} n \\ i \end{pmatrix} e(kT_s + (n-i)T)}_{e_f(kT_s+nT)}, \tag{16}$$

where x_f and e_f are referred to as noise-free filtered output and filtered noise, respectively, and they are stacked in the vectors

$$\boldsymbol{X}_f = \begin{bmatrix} x_f(nT) \\ x_f(T_s + nT) \\ \vdots \\ x_f((N-1)T) \end{bmatrix} \in \mathbb{R}^{\tilde{N}}, \quad \boldsymbol{E}_f = \begin{bmatrix} e_f(nT) \\ e_f(T_s + nT) \\ \vdots \\ e_f((N-1)T) \end{bmatrix} \in \mathbb{R}^{\tilde{N}}. \tag{17}$$

Let us define the noise-free regressor $\boldsymbol{\phi}^{\mathrm{o}}(kT_s + nT)$ similarly to (7), using the noise-free sampled output $x(kT_s)$ instead of $y(kT_s)$, *i.e.*,

$$\boldsymbol{\phi}^{\mathrm{o}}(kT_s + nT) = \begin{bmatrix} -T_s \sum_{k_n=0}^{N_p-1} \sum_{i=0}^{n-1} (-1)^i \begin{pmatrix} n-1 \\ i \end{pmatrix} x(kT_s + k_nT_s + (n-1-i)T) \\ -T_s^2 \sum_{k_n=0}^{N_p-1} \sum_{k_{n-1}=0}^{N_p-1} \sum_{i=0}^{n-2} (-1)^i \begin{pmatrix} n-2 \\ i \end{pmatrix} x(kT_s + k_nT_s + k_{n-1}T_s + (n-2-i)T) \\ \vdots \\ -T_s^n \sum_{k_n=0}^{N_p-1} \sum_{k_{n-1}=0}^{N_p-1} \cdots \sum_{k_1=0}^{N_p-1} x(kT_s + k_nT_s + k_{n-1}T_s + \cdots k_1T_s) \\ T_s \sum_{k_n=0}^{N_p-1} \sum_{i=0}^{n-1} (-1)^i \begin{pmatrix} n-1 \\ i \end{pmatrix} u(kT_s + k_nT_s + (n-1-i)T) \\ \vdots \\ T_s^n \sum_{k_n=0}^{N_p-1} \sum_{k_{n-1}=0}^{N_p-1} \cdots \sum_{k_1=0}^{N_p-1} u(kT_s + k_nT_s + k_{n-1}T_s + \cdots k_1T_s) \end{bmatrix}, \tag{18}$$

and let $\boldsymbol{\Phi}^{\mathrm{o}}$ be the corresponding noise-free regressor matrix defined as

$$\boldsymbol{\Phi}^{\mathrm{o}} = \begin{bmatrix} \boldsymbol{\phi}^{\mathrm{o}}(nT)^T \\ \boldsymbol{\phi}^{\mathrm{o}}(T_s + nT)^T \\ \vdots \\ \boldsymbol{\phi}^{\mathrm{o}}((N-1)T_s)^T \end{bmatrix} \in \mathbb{R}^{\tilde{N},2n}. \tag{19}$$

Based on the matrix definition above, the filtered noise-free output vector X_f is given by

$$\boldsymbol{X}_f = \boldsymbol{\Phi}^{\mathrm{o}}\boldsymbol{\theta}^{\mathrm{o}} + \boldsymbol{V}, \tag{20}$$

where $\boldsymbol{V}$ is the truncation error due to the forward-Euler approximation of the integrals in (3) in the construction of the noise-free regressor $\boldsymbol{\phi}^{\mathrm{o}}(kT_s + nT)$. Thus, the filtered output vector $\boldsymbol{Y}_f$ is factorized as $\boldsymbol{Y}_f = \boldsymbol{\Phi}^{\mathrm{o}}\boldsymbol{\theta}^{\mathrm{o}} + \boldsymbol{E}_f + \boldsymbol{V}$ and the difference between the estimated parameters $\hat{\boldsymbol{\theta}}_{\mathrm{LS}}$ and the true parameters $\boldsymbol{\theta}^{\mathrm{o}}$ can be expressed as follows

$$\begin{aligned} \hat{\boldsymbol{\theta}}_{\mathrm{LS}} - \boldsymbol{\theta}^{\mathrm{o}} &= \left(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\boldsymbol{Y}_f - \boldsymbol{\theta}^{\mathrm{o}} = \left(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\left(\boldsymbol{\Phi}^{\mathrm{o}}\boldsymbol{\theta}^{\mathrm{o}} + \boldsymbol{E}_f + \boldsymbol{V}\right) - \boldsymbol{\theta}^{\mathrm{o}} = \\ &= \underbrace{\left(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\boldsymbol{\Delta}\boldsymbol{\Phi}^{\mathrm{o}}\boldsymbol{\theta}^{\mathrm{o}}}_{\boldsymbol{B}_{\boldsymbol{\Delta\Phi}}} + \underbrace{\left(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\boldsymbol{E}_f}_{\boldsymbol{B}_{\boldsymbol{E}}} + \underbrace{\left(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\boldsymbol{V}}_{\boldsymbol{B}_{\boldsymbol{V}}}, \end{aligned} \tag{21}$$

with $\boldsymbol{\Delta\Phi}^{\mathrm{o}} = \boldsymbol{\Phi}^{\mathrm{o}} - \boldsymbol{\Phi}$. Note that $\boldsymbol{B}_{\boldsymbol{V}}$ is the error due to the forward-Euler approximation and it

can be reduced by increasing the sampling frequency. The error terms $\boldsymbol{B}_{\boldsymbol{\Delta\Phi}}$ and $\boldsymbol{B}_{\boldsymbol{E}}$ are due to the noise $e(kT_s)$ corrupting the output observation $y(kT_s)$. More specifically, $\boldsymbol{B}_{\boldsymbol{\Delta\Phi}}$ is due to the fact that noisy outputs are used to build the regressor matrix $\boldsymbol{\Phi}$ (thus $\boldsymbol{\Delta\Phi}^{\mathrm{o}} \neq 0$), while $\boldsymbol{B}_{\boldsymbol{E}}$ is caused by the noise $e_f(kT_s + nT)$ corrupting the filtered output $y_f(kT_s + nT)$.

Since the output noise $e(kT_s)$ is assumed to be generated by a zero-mean white process independent of the input $u(kT_s)$ and of the noise-free output $x(kT_s)$, the following limits hold *with probability* 1 (w.p. 1) under the assumption that $u(t)$ and $x(t)$ are bounded (Ljung, 1999):

$$\lim_{\tilde{N}\to\infty} \frac{1}{\tilde{N}} \sum_{k=0}^{\tilde{N}} \sum_{k'=0}^{\tilde{N}} e(kT_s)e(k'T_s) = \lim_{\tilde{N}\to\infty} \frac{1}{\tilde{N}} \sum_{k=0}^{\tilde{N}} e^2(kT_s) = \mathbb{E}\left[e^2(kT_s)\right] = \sigma_e^2, \tag{22a}$$

$$\lim_{\tilde{N}\to\infty} \frac{1}{\tilde{N}} \sum_{k=0}^{\tilde{N}} x(kT_s)e(kT_s) = \mathbb{E}\left[x(kT_s)e(kT_s)\right] = 0, \tag{22b}$$

$$\lim_{\tilde{N}\to\infty} \frac{1}{\tilde{N}} \sum_{k=0}^{\tilde{N}} u(kT_s)e(kT_s) = \mathbb{E}\left[u(kT_s)e(kT_s)\right] = 0. \tag{22c}$$

Using the asymptotic results in (22), the asymptotic bias in the least-squares estimate can be computed and it is equal (w.p. 1) to

$$\lim_{\tilde{N}\to\infty} \hat{\boldsymbol{\theta}}_{\mathrm{LS}} - \boldsymbol{\theta}^{\mathrm{o}} = \underbrace{\boldsymbol{\Gamma}_* \boldsymbol{Q}_1 \boldsymbol{\theta}^{\mathrm{o}} \sigma_e^2}_{\lim_{\tilde{N}\to\infty} \boldsymbol{B}_{\boldsymbol{\Delta\Phi}}} + \underbrace{\boldsymbol{\Gamma}_* \boldsymbol{Q}_2 \sigma_e^2}_{\lim_{\tilde{N}\to\infty} \boldsymbol{B}_{\boldsymbol{E}}} + \underbrace{\boldsymbol{\Gamma}_* \boldsymbol{\Phi}^T \boldsymbol{V}}_{\lim_{\tilde{N}\to\infty} \boldsymbol{B}_{\boldsymbol{V}}}, \tag{23}$$

where

$$\boldsymbol{\Gamma}_* = \lim_{\tilde{N}\to\infty} \boldsymbol{\Gamma}, \text{ with } \boldsymbol{\Gamma} = \left(\frac{1}{\tilde{N}} \boldsymbol{\Phi}^T \boldsymbol{\Phi}\right)^{-1}, \tag{24}$$

and $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ are properly defined matrices obtained by computing the expression of $\frac{1}{\tilde{N}}\boldsymbol{\Phi}^T\boldsymbol{\Delta\Phi}^{\mathrm{o}}$ and $\frac{1}{\tilde{N}}\boldsymbol{\Phi}^T\boldsymbol{E}_f$, respectively, and substituting the limits in (22). An illustrative example reporting the values of the matrices $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ is presented in Section 4.

Although the asymptotic values of the errors $\boldsymbol{B}_{\boldsymbol{\Delta\Phi}}$ and $\boldsymbol{B}_{\boldsymbol{E}}$ caused by the noise $e(kT_s)$ cannot be computed because of their dependence on the true system parameters $\boldsymbol{\theta}^{\mathrm{o}}$ and on the noise variance σ_e^2 (which is often unknown), eq. (23) provides useful information to derive the following asymptotic upper bounds on the errors $\boldsymbol{B}_{\boldsymbol{\Delta\Phi}}$ and $\boldsymbol{B}_{\boldsymbol{E}}$:

$$\frac{\lim_{\tilde{N}\to\infty} \|\boldsymbol{B}_{\boldsymbol{\Delta\Phi}}\|}{\|\boldsymbol{\theta}^{\mathrm{o}}\|} \leq \|\boldsymbol{\Gamma}_* \boldsymbol{Q}_1\| \, \sigma_e^2 = \lim_{\tilde{N}\to\infty} \|\boldsymbol{\Gamma}\boldsymbol{Q}_1\| \, \sigma_e^2, \tag{25a}$$

$$\lim_{\tilde{N}\to\infty} \|\boldsymbol{B}_{\boldsymbol{E}}\| \leq \|\boldsymbol{\Gamma}_* \boldsymbol{Q}_2\| \, \sigma_e^2 = \lim_{\tilde{N}\to\infty} \|\boldsymbol{\Gamma}\boldsymbol{Q}_2\| \, \sigma_e^2, \tag{25b}$$

where the right sides of the equations hold because of continuity of the norm operator and definition of the matrix $\boldsymbol{\Gamma}_*$.

Thus, $\|\boldsymbol{\Gamma}\boldsymbol{Q}_1\|$ provides an asymptotic upper bound on the ratio between the norm of the relative bias term $\frac{\|\boldsymbol{B}_{\boldsymbol{\Delta\Phi}}\|}{\|\boldsymbol{\theta}^{\mathrm{o}}\|}$ and the noise variance σ_e^2. Similarly, $\|\boldsymbol{\Gamma}\boldsymbol{Q}_2\|$ provides an asymptotic upper bound on the ratio between the norm of the bias term $\|\boldsymbol{B}_{\boldsymbol{E}}\|$ and σ_e^2. Thus, $\|\boldsymbol{\Gamma}\boldsymbol{Q}_1\|$ and $\|\boldsymbol{\Gamma}\boldsymbol{Q}_2\|$ can be used as criteria to select the integration horizon N_p based on sampled data, as described in the following example.

4. Simulation examples

4.1 *Illustrative example*

For illustrative purposes, an example on the identification of a first-order continuous-time system is now discussed. Through this simple example, we also provide an intuitive explanation about the effect of the integration horizon N_p on the bias affecting the least-squares estimator $\hat{\boldsymbol{\theta}}_{\text{LS}}$.

4.1.1 System description

Let us consider the system $\mathcal{S}$ described by the first-order ODE

$$x^{(1)}(t) = -a_0^{\text{o}} x(t) + b_0^{\text{o}} u(t), \tag{26}$$

whose corresponding output predictor in (6) has the form

$$\hat{y}(kT_s + T) = y(kT_s) - a_0 T_s \sum_{k_1=0}^{N_p-1} y(kT_s + k_1 T_s) + b_0 T_s \sum_{k_1=0}^{N_p-1} u(kT_s + k_1 T_s). \tag{27}$$

4.1.2 Definitions and approximations

The noise-free $\boldsymbol{\phi}^{\text{o}}(kT_s + T)$ and noisy regressor $\boldsymbol{\phi}(kT_s + T)$ associated to the considered system are

$$\boldsymbol{\phi}^{\text{o}}(kT_s + T) = \begin{bmatrix} -T_s \sum\limits_{k_1=0}^{N_p-1} x(kT_s + k_1 T_s) \\ T_s \sum\limits_{k_1=0}^{N_p-1} u(kT_s + k_1 T_s) \end{bmatrix}, \tag{28a}$$

$$\boldsymbol{\phi}(kT_s + T) = \begin{bmatrix} -T_s \sum\limits_{k_1=0}^{N_p-1} y(kT_s + k_1 T_s) \\ T_s \sum\limits_{k_1=0}^{N_p-1} u(kT_s + k_1 T_s) \end{bmatrix}, \tag{28b}$$

and each row of the matrix $\boldsymbol{\Delta\Phi}^{\text{o}}$ is thus given by

$$\boldsymbol{\Delta\phi}^{\text{o}}(kT_s + T)^T = \boldsymbol{\phi}^{\text{o}}(kT_s + T)^T - \boldsymbol{\phi}(kT_s + T)^T = \begin{bmatrix} T_s \sum\limits_{k_1=0}^{N_p-1} e(kT_s + k_1 T_s) & 0 \end{bmatrix}. \tag{28c}$$

Based on the definition of the regressor $\boldsymbol{\phi}(kT_s + T)$ in (28b), the matrix $\boldsymbol{\Phi}^T\boldsymbol{\Phi}$ is

$$\boldsymbol{\Phi}^T\boldsymbol{\Phi} = \sum_{k=0}^{\tilde{N}} T_s^2 \begin{bmatrix} \sum\limits_{k_1=0}^{N_p-1} y(kT_s + k_1 T_s) \sum\limits_{k_1=0}^{N_p-1} y(kT_s + k_1 T_s) & -\sum\limits_{k_1=0}^{N_p-1} y(kT_s + k_1 T_s) \sum\limits_{k_1=0}^{N_p-1} u(kT_s + k_1 T_s) \\ -\sum\limits_{k_1=0}^{N_p-1} y(kT_s + k_1 T_s) \sum\limits_{k_1=0}^{N_p-1} u(kT_s + k_1 T_s) & \sum\limits_{k_1=0}^{N_p-1} u(kT_s + k_1 T_s) \sum\limits_{k_1=0}^{N_p-1} u(kT_s + k_1 T_s) \end{bmatrix}, \tag{29}$$

where the operator $\sum\limits_{k=0}^{\tilde{N}} T_s^2$ is applied element-wise.

Let us assume that the signals $x(t)$ and $u(t)$ does not have a significant variation over the integration interval $[t\ \ t{+}T]$, so that $\sum_{k_1=0}^{N_p-1} y(kT_s{+}k_1T_s)$ and $\sum_{k_1=0}^{N_p-1} u(kT_s{+}k_1T_s)$ can be approximated as

$$\sum_{k_1=0}^{N_p-1} y(kT_s+k_1T_s) \approx N_p x(kT_s) + \sum_{k_1=0}^{N_p-1} e(kT_s+k_1T_s), \tag{30a}$$

$$\sum_{k_1=0}^{N_p-1} u(kT_s+k_1T_s) \approx N_p u(kT_s). \tag{30b}$$

Note that the approximations (30) might not be reasonable anymore for large values of N_p. By substituting (30) into (29) and using properties (22), we obtain

$$\boldsymbol{\Gamma}_*^{-1} = \lim_{\tilde{N}\to\infty} \frac{1}{\tilde{N}} \boldsymbol{\Phi}^T \boldsymbol{\Phi} \approx \lim_{\tilde{N}\to\infty} T_s^2 \begin{bmatrix} N_p\sigma_e^2 + \frac{1}{\tilde{N}} N_p^2 \sum_{k=0}^{\tilde{N}} x^2(kT_s) & -\frac{1}{\tilde{N}} N_p^2 \sum_{k=0}^{\tilde{N}} x(kT_s)u(kT_s) \\ -\frac{1}{\tilde{N}} N_p^2 \sum_{k=0}^{\tilde{N}} x(kT_s)u(kT_s) & \frac{1}{\tilde{N}} N_p^2 \sum_{k=0}^{\tilde{N}} u^2(kT_s) \end{bmatrix}. \tag{31}$$

Under the assumption of high signal-to-noise-ratio (namely, for $\frac{1}{\tilde{N}} \sum_{k=0}^{\tilde{N}} x^2(kT_s) \gg \sigma_e^2$), the matrix $\boldsymbol{\Gamma}_*^{-1}$ is further approximated as follows:

$$\boldsymbol{\Gamma}_*^{-1} \approx \lim_{\tilde{N}\to\infty} T_s^2 \begin{bmatrix} \frac{1}{\tilde{N}} N_p^2 \sum_{k=0}^{\tilde{N}} x^2(kT_s) & -\frac{1}{\tilde{N}} N_p^2 \sum_{k=0}^{\tilde{N}} x(kT_s)u(kT_s) \\ -\frac{1}{\tilde{N}} N_p^2 \sum_{k=0}^{\tilde{N}} x(kT_s)u(kT_s) & \frac{1}{\tilde{N}} N_p^2 \sum_{k=0}^{\tilde{N}} u^2(kT_s) \end{bmatrix}. \tag{32}$$

4.1.3 Asymptotic bias $\boldsymbol{B}_{\boldsymbol{\Delta\Phi}}$

Let us focus on the computation of the asymptotic value of the error term $\boldsymbol{B}_{\boldsymbol{\Delta\Phi}}$ in (21). Using the expression of $\boldsymbol{\Delta}\boldsymbol{\phi}^{\mathrm{o}}(kT_s+T)$ in (28c), the matrix $\boldsymbol{\Phi}^T\boldsymbol{\Delta}\boldsymbol{\Phi}^{\mathrm{o}}$ can be computed and it is equal to

$$\boldsymbol{\Phi}^T\boldsymbol{\Delta}\boldsymbol{\Phi}^{\mathrm{o}} = \begin{bmatrix} -T_s^2 \sum_{k=0}^{\tilde{N}} \left[\sum_{k_1=0}^{N_p-1} y(kT_s+k_1T_s) \sum_{k_1=0}^{N_p-1} e(kT_s+k_1T_s) \right] & 0 \\ -T_s^2 \sum_{k=0}^{\tilde{N}} \left[\sum_{k_1=0}^{N_p-1} u(kT_s+k_1T_s) \sum_{k_1=0}^{N_p-1} e(kT_s+k_1T_s) \right] & 0 \end{bmatrix}. \tag{33}$$

Thus, from properties (22), the following limit holds w.p. 1:

$$\lim_{\tilde{N}\to\infty} \frac{1}{\tilde{N}} \boldsymbol{\Phi}^T\boldsymbol{\Delta}\boldsymbol{\Phi}^{\mathrm{o}} = \underbrace{\begin{bmatrix} -T_s^2 N_p & 0 \\ 0 & 0 \end{bmatrix}}_{\boldsymbol{Q}_1} \sigma_e^2. \tag{34}$$

By using the approximation of $\boldsymbol{\Gamma}_*^{-1}$ in (32), the asymptotic bias $\lim_{\tilde{N}\to\infty} \boldsymbol{B}_{\boldsymbol{\Delta\Phi}}$ in (23) can be

approximated as follows

$$\lim_{\tilde{N}\to\infty} \boldsymbol{B}_{\boldsymbol{\Delta\Phi}} = \boldsymbol{\Gamma}_* \boldsymbol{Q}_1 \boldsymbol{\theta}^{\mathrm{o}} \sigma_e^2 \approx \frac{1}{N_p} \lim_{\tilde{N}\to\infty} \begin{bmatrix} \frac{1}{\tilde{N}} \sum\limits_{k=0}^{\tilde{N}} x^2(kT_s) & -\frac{1}{\tilde{N}} \sum\limits_{k=0}^{\tilde{N}} x(kT_s)u(kT_s) \\ -\frac{1}{\tilde{N}} \sum\limits_{k=0}^{\tilde{N}} x(kT_s)u(kT_s) & \frac{1}{\tilde{N}} \sum\limits_{k=0}^{\tilde{N}} u^2(kT_s) \end{bmatrix}^{-1} \begin{bmatrix} -1 & 0 \\ 0 & 0 \end{bmatrix} \boldsymbol{\theta}^{\mathrm{o}} \sigma_e^2. \tag{35}$$

Thus, the asymptotic bias $\lim\limits_{\tilde{N}\to\infty} \boldsymbol{B}_{\boldsymbol{\Delta\Phi}}$ decreases with a factor $\frac{1}{N_p}$. However, we remind that the approximation of $\boldsymbol{\Gamma}_*$ in (32) (and consequently the approximation in (35)) may not be reasonable for large values of N_p.

4.1.4 Asymptotic bias $\boldsymbol{B}_{\boldsymbol{E}}$

Let us now compute the asymptotic value of the error term $\boldsymbol{B}_{\boldsymbol{E}}$. Since the filtered noise is

$$e_f(kT_s + T) = e(kT_s + T) - e(kT_s), \tag{36}$$

the vector $\boldsymbol{\Phi}^T \boldsymbol{E}_f$ is equal to

$$\boldsymbol{\Phi}^T \boldsymbol{E}_f = \begin{bmatrix} -T_s \sum\limits_{k=0}^{\tilde{N}} \left[\sum\limits_{k_1=0}^{N_p-1} y(kT_s + k_1T_s) \left(e(kT_s + T) - e(kT_s) \right) \right] \\ 0 \end{bmatrix}. \tag{37}$$

Thus, from properties (22), we have:

$$\begin{aligned} &\lim_{\tilde{N}\to\infty} \frac{1}{\tilde{N}} \boldsymbol{\Phi}^T \boldsymbol{E}_f = \\ &= \lim_{\tilde{N}\to\infty} \frac{1}{\tilde{N}} \begin{bmatrix} -T_s \sum\limits_{k=0}^{\tilde{N}} \left[\sum\limits_{k_1=0}^{N_p-1} \left(x(kT_s + k_1T_s) + e(kT_s + k_1T_s) \right) \left(e(kT_s + T) - e(kT_s) \right) \right] \\ 0 \end{bmatrix} = \\ &= \lim_{\tilde{N}\to\infty} \frac{1}{\tilde{N}} \begin{bmatrix} T_s \sum\limits_{k=0}^{\tilde{N}} e^2(kT_s) \\ 0 \end{bmatrix} = \underbrace{\begin{bmatrix} T_s \\ 0 \end{bmatrix}}_{\boldsymbol{Q}_2} \sigma_e^2, \end{aligned}$$

where the limit holds w.p. 1. Using the approximation of $\boldsymbol{\Gamma}_*^{-1}$ in (32), the error term $\boldsymbol{B}_{\boldsymbol{E}}$ in (23) can be asymptotically approximated by

$$\lim_{\tilde{N}\to\infty} \boldsymbol{B}_{\boldsymbol{E}} = \boldsymbol{\Gamma}_* \boldsymbol{Q}_2 \sigma_e^2 \approx \frac{T_s}{N_p^2} \lim_{\tilde{N}\to\infty} \begin{bmatrix} \frac{1}{\tilde{N}} \sum\limits_{k=0}^{\tilde{N}} x^2(kT_s) & -\frac{1}{\tilde{N}} \sum\limits_{k=0}^{\tilde{N}} x(kT_s)u(kT_s) \\ -\frac{1}{\tilde{N}} \sum\limits_{k=0}^{\tilde{N}} x(kT_s)u(kT_s) & \frac{1}{\tilde{N}} \sum\limits_{k=0}^{\tilde{N}} u^2(kT_s) \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \sigma_e^2. \tag{38}$$

Thus, the bias $\lim\limits_{\tilde{N}\to\infty} \boldsymbol{B}_{\boldsymbol{E}}$ descreases with a factor $\frac{1}{N_p^2}$.

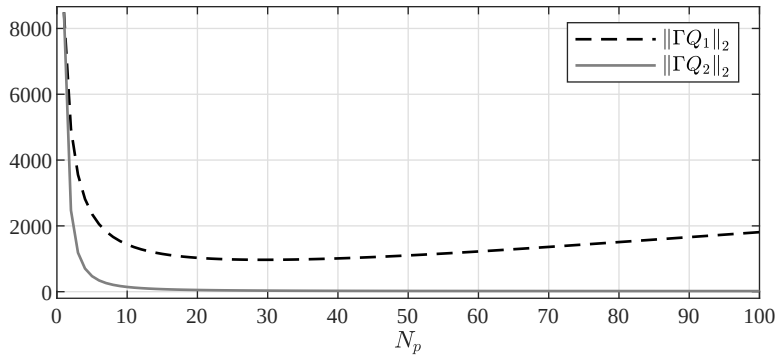


Figure 1. Illustrative example: 2-norms $\|\boldsymbol{\Gamma Q}_1\|_2$ and $\|\boldsymbol{\Gamma Q}_2\|_2$ vs integration horizon N_p.

4.1.5 Numerical results

In order to show some numerical results, a set of data is generated from the ODE (26) for parameters $a_0^{\mathrm{o}} = 9$ and $b_0^{\mathrm{o}} = -0.3$. Input and output samples are measured with a sampling interval $T_s = 0.01$. The input inter-sample behaviour is of *zero-order-hold* (ZOH) type, *i.e.*,

$$u(t) = u(kT_s), \quad \text{for } t \in [kT_s \;\; kT_s + T_s], \tag{39}$$

where $u(kT_s)$, $k = 0, 1, \ldots$, is generated by a DT zero-mean Gaussian white noise process with standard deviation $\sigma_u = 2$ followed by a discrete-time filter with z-transfer function

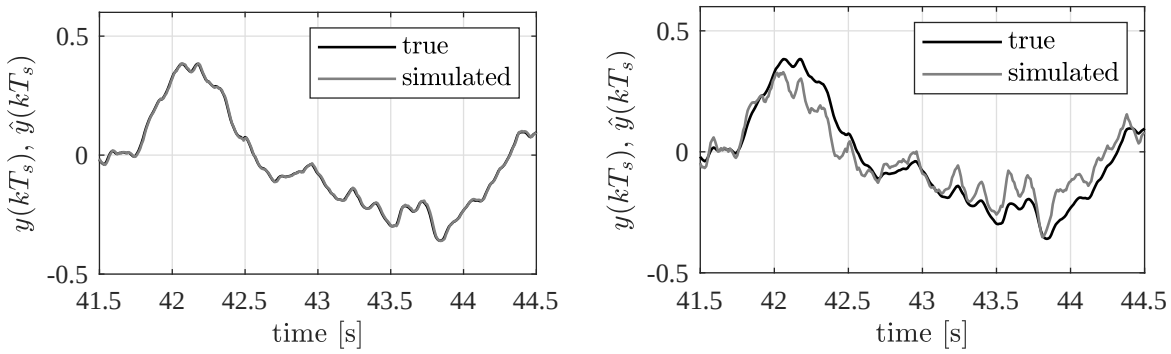
$$F_u(z) = \frac{1}{z - 0.94}. \tag{40}$$

The noise $e(kT_s)$ corrupting the output observation is a zero-mean Gaussian random variable with standard deviation $\sigma_e = 0.05$. This corresponds to a *signal-to-noise ratio*

$$\mathrm{SNR} = 10 \log_{10} \frac{\sum_{k=0}^{N-1} x^2(kT_s)}{\sum_{k=0}^{N-1} e^2(kT_s)} = 9.6 \text{ dB}.$$

A dataset with $N = 500,000$ input and output samples is generated and used to estimate the parameters a_0^{o} and b_0^{o} through the numerical-integration approach described in Section 3. The 2-norms $\|\boldsymbol{\Gamma Q}_1\|_2$ and $\|\boldsymbol{\Gamma Q}_2\|_2$ are computed and plotted in Figure 1 for different values of the integration horizon N_p. We remind that, according to eq. (25), $\|\boldsymbol{\Gamma Q}_1\|$ and $\|\boldsymbol{\Gamma Q}_2\|$ provide asymptotic upper bounds (up to the constant σ_e^2) on the norms $\frac{\|\boldsymbol{B}_{\boldsymbol{\Delta\Phi}}\|}{\|\boldsymbol{\theta}^{\mathrm{o}}\|}$ and $\|\boldsymbol{B}_{\boldsymbol{E}}\|$, respectively. As discussed in Sections 4.1.3 and 4.1.4, for $\tilde{N} \to \infty$, the norms $\|\boldsymbol{\Gamma Q}_1\|$ and $\|\boldsymbol{\Gamma Q}_2\|$ decreases with factors $\frac{1}{N_p}$ and $\frac{1}{N_p^2}$, respectively. However, a slight increase of $\|\boldsymbol{\Gamma Q}_1\|_2$ can be observed in Figure 1 for values of N_p greater than 30. This is not surprising since the approximations in (30) and (31) may not be reasonable for large integration horizons.

The value $N_p = 30$ providing the minimum of the sum $\|\boldsymbol{\Gamma Q}_1\|_2 \, \|\hat{\boldsymbol{\theta}}_{\mathrm{LS}}(N_p)\|_2 + \|\boldsymbol{\Gamma Q}_2\|_2$ is chosen, with corresponding least-squares estimator $\hat{\boldsymbol{\theta}}_{\mathrm{LS}} = [8.923 \quad -0.295]$. The performance of the estimated model is assessed on noise-free test data (not used for training) of length $N_{\mathrm{v}} = 10,000$. A subsequence of test outputs $y(kT_s)$ is plotted in Figure 2(a), along with the simulated output $\hat{y}(kT_s)$ of the model. Note that the plots of true and simulated output are almost overlapped in Figure 2(a) and thus they cannot be distinguished. The fit between simulated and true ouput is



(a) $N_p = 30$ (b) $N_p = 1$

Figure 2. Illustrative example. Actual output $y(kT_s)$ (gray line) and simulated output $\hat{y}(kT_s)$ (black line) provided by the model obtained for integration horizon $N_p = 30$ (left panel) and for $N_p = 1$ (right panel). Gray and black lines are almost overlapped in the left panel, and thus they cannot be distinguished.

measured by the *Best Fit Rate* (BFR) defined as

$$\text{BFR} = 1 - \sqrt{\frac{\sum_{k=0}^{N_\text{v}-1} (y(kT_s) - \hat{y}(kT_s))^2}{\sum_{k=0}^{N_\text{v}-1} (y(kT_s) - \bar{y})^2}},$$

where $\bar{y} = \frac{1}{N_\text{v}} \sum_{k=0}^{N_\text{v}-1} y(kT_s)$ is the sample mean of the output. The achieved BFR is 0.97, which confirms a good match between true and simulated output.

For the sake of comparison, the performance of the model obtained for $N_p = 1$ is evaluated. As discussed in Remark 1, when $T = T_s$ (or equivalently $N_p = 1$), the integration-based identification method is equivalent to the identification of the DT model derived from the discretization of the ODE (26). The estimated parameters are $\hat{\boldsymbol{\theta}}_\text{LS} = [28.135 \quad -0.664]$, significantly different than the true ones. A poor estimate of the system parameters is reflected on the simulated output of the model (plotted in Figure 2(b)) and on the achieved BFR = 0.61. A comparison of the results obtained for $N_p = 30$ and $N_p = 1$ shows the advantages of the presented CT identification method w.r.t. an indirect approach based on the identification of the discretized model.

4.1.6 Monte Carlo simulation

A Monte Carlo simulation of 100 realizations is carried out to show the behaviour of the CT identification algorithm w.r.t. different realizations of training data.

At each Monte Carlo run, the integration horizon N_p minimizing (the approximation of) the bias upper bound $\|\boldsymbol{\Gamma Q}_1\|_2 \, \|\hat{\boldsymbol{\theta}}_\text{LS}(N_p)\|_2 + \|\boldsymbol{\Gamma Q}_2\|_2$ is chosen. Among the 100 runs, $N_p = 29$ and $N_p = 30$ is selected 4 and 96 times, respectively. Monte Carlo simulation results are summarized in Table 1, which shows the mean and the standard deviation of the estimated parameters a_0 and b_0, along with the BFR achieved on validation data. It can be observed that, although a small bias is still present, the parameters' estimates are affected by a very small variance. For the sake of comparison, results obtained by the LS-based DT identification algorithm (*i.e.*, for $N_p = 1$) are also reported.

Table 1. Summary of Monte Carlo simulation: mean and standard deviation of the parameters' estimates and of the achieved BFR.

		True	CT identification	DT identification
a_0	mean	9	8.9242	28.1170
	std	–	0.0118	0.1302
b_0	mean	−0.3	−0.2953	−0.6638
	std	–	0.0003	0.0022
BFR	mean	–	0.9687	0.5410
	std	–	0.0001	0.0027

4.2 *Rao-Garnier benchmark*

In order to further show the benefits of numerical integration w.r.t. indirect identification of a discretized model, we consider the Rao-Garnier benchmark (Rao & Garnier, 2002), which addresses the identification of the forth-order non-minimum phase LTI system with Laplace transfer function

$$G(s) = \frac{-T_c s + 1}{\left(\frac{s^2}{\omega_{n,1}^2} + \frac{2\zeta_1 s}{\omega_{n,1}} + 1\right)\left(\frac{s^2}{\omega_{n,2}^2} + \frac{2\zeta_2 s}{\omega_{n,2}} + 1\right)}, \tag{41}$$

with $T_c = 4$ s, $\omega_{n,1} = 20$ rad/s, $\zeta_1 = 0.1$, $\omega_{n,2} = 2$ rad/s, and $\zeta_2 = 0.25$.

A Monte Carlo simulation of 100 random realizations is carried out. At each Monte Carlo run, the output observations are sampled uniformly every 10 ms, and the system is simulated for 500 s using a PRBS input signal under a zero-order hold assumption. The output measurements are corrupted by an additive Gaussian error $e(kT_s)$ generated by a white noise process with zero mean and standard deviation $\sigma_e = 0.15$. This corresponds to an average SNR equal to 19 dB.

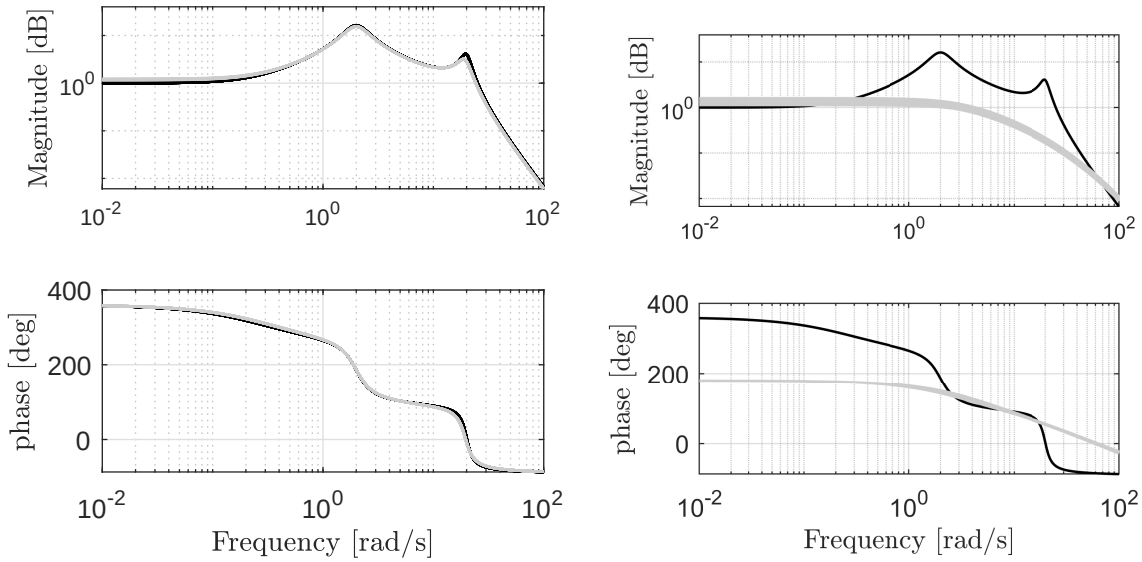
Both continuous-time identification (with integration horizon $N_p = 26$) and discrete-time identification is performed. The Bode plots of the estimated models are reported in Figure 3, which clearly show the benefit of numerical integration. As in the numerical example discussed in the previous paragraph, the estimate of the Bode diagram is affected by a very small variance, although an unavoidable bias in the model estimate (caused by the use of a least-squares estimator) is still present. The open-loop simulated output, in a validation dataset, provided by the model estimated in the first Monte Carlo run is plotted in Figure 4. This figure shows, for the case of CT identification, a good match between actual and simulated output. On the other hand, poor results are achieved by indirect identification of the discretized model.

5. Extension to non-linear system identification and experimental case study

The numerical-integration based approach presented in the paper is now extended to the identification of nonlinear systems and applied to estimate the torque-displacement characteristic of *qb-move advance* (Catalano et al., 2011; Della Santina et al., 2017), a *variable stiffness actuator* (VSA) shown in Figure 5(a) and produced by *qb-robotics s.r.l.*, an Italian company which develops innovative devices for soft-robotics technology. Besides showing the effectiveness of the proposed approach in a real example, the aim of this case study is to show the benefits of numerical integration even in the identification of nonlinear systems based on data collected from closed-loop experiments.

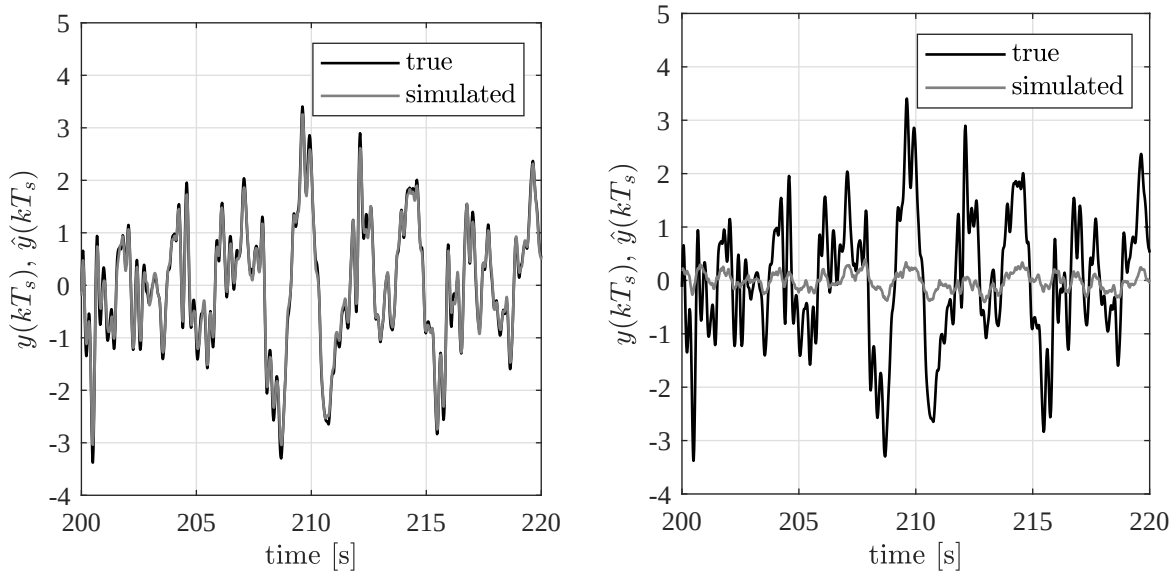
5.1 *Experimental setup*

The *qb-move advance* actuator implements an agonist-antagonist mechanism sketched in Figure 5(b), where two internal motors are connected to the output shaft via a set of tensioning



(a) CT identification ($N_p = 26$) (b) DT identification ($N_p = 1$)

Figure 3. Rao-Garnier benchmark: true Bode diagram (black line) vs estimated Bode diagrams (gray lines).



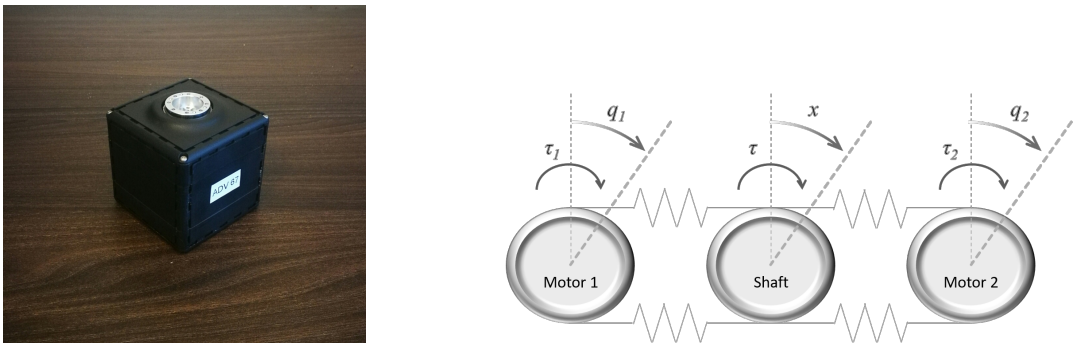
(a) CT identitifcation ($N_p = 26$) (b) DT identitifcation ($N_p = 1$)

Figure 4. Rao-Garnier benchmark: actual validation output and open-loop simulated output estimated in the first Monte Carlo run.

mechanisms controlled by servomotors. Both motors can be used to influence either the overall compliance of the actuator or the equilibrium position, which is given by

$$x_{\text{eq}} = \frac{q_1 + q_2}{2}, \tag{42}$$

where q_1 and q_2 are the positions of the internal motors. The overall torque τ of the actuator is given by the following nonlinear function of the shaft position x and of the positions q_1 and q_2 of



(a) (b)

Figure 5. *qb-move advance*: (a) exterior of the actuator; (b) schematic of the implemented agonist-antagonistic mechanism.

the internal motors:

$$\tau = r_1 \tan\left(a_1(x - q_1 - b_1)\right) + r_2 \tan\left(a_2(x - q_2 - b_2)\right), \tag{43a}$$

where the parameters $r_1, r_2, a_1, a_2, b_1, b_2$ characterizing the VSA are unknown and have to be estimated from data.

The experimental setup used to gather data is visualized in Figure 6 and it consists of a one-degree-of-freedom pendulum actuated by *qb-move advanced*, with a mass of weight $m = 1$ kg mounted on a link of length $l = 0.17$ m and negligible weight. The behaviour of the overall system is thus described by the second-order nonlinear differential equation

$$x^{(2)}(t) + \tfrac{c}{ml} x^{(1)}(t) + g \sin(x(t)) = \tfrac{1}{ml}\tau(t), \tag{43b}$$

where $g = 9.81\ m/s^2$ is the gravity acceleration, c is an unknown damping coefficient and τ is the torque applied by the actuator, which is not directly measured and described by the nonlinear relation (43a).

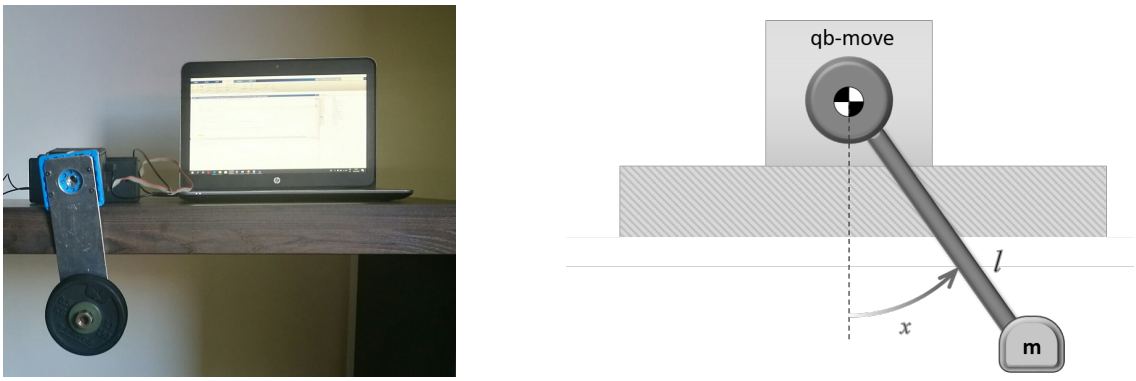
In order to gather data, a closed-loop experiment is performed using a multisine trajectory as desired position for the link. Five different values of stiffness' actuator are set during the experiment. The actual link's position x and the internal motors' positions q_1 and q_2 are measured with a sampling time $T_s = 1$ ms, and the total duration of the experiment is 250 seconds. Thus, $250,000$ samples $\{q_1(kT_s), q_2(kT_s), y(kT_s)\}$ are gathered, with $y(kT_s)$ being the (noisy) measurement of the link's position x at time kT_s. The first $N = 200,000$ samples are used to estimate the unknown parameters $r_1, r_2, a_1, a_2, b_1, b_2$ characterizing the VSA (eq. (43a)), while the remaining $N_\mathrm{v} = 50,000$ samples are used for validation.

5.2 *Identification algorithm*

Note that the CT identification approach described in Section 3 cannot be directly applied for the estimation of the parameters of the ODE (43) because of:

- nonlinear dependence of the ODE (43b) on the output signal $x(t)$;
- nonlinear dependence of τ (eq. (43)) on the parameters a_1, a_2, b_1 and b_2.

Although the first issue can be easily handled by taking into account the nonlinear dependence on $x(t)$ in the construction of the regressor $\boldsymbol{\phi}$, the second issue does not allow us to solve the least-squares problem analytically as in (14). Therefore, the hybrid approach described in the following,



(a) Test bench (b) Schematic representation

Figure 6. Experimental setup.

that combines linear regression and particle swarm optimization, is used to estimate the unknown parameters of the ODE (43).

First, according to the numerical-integration based method described in the paper, the ODE (43) is integrated twice over a time window of length T, thus obtaining

$$\begin{aligned} & x(t_2+2T) - 2x(t_2+T) + x(t_2) = \\ = & -\frac{c}{ml}\int_{t_2}^{t_2+T}(x(t+T)-x(t))dt - g\int_{t_2}^{t_2+T}\int_{t_1}^{t_1+T}\sin(x(t))dtdt_1 + \\ & + \frac{r_1}{ml}\int_{t_2}^{t_2+T}\int_{t_1}^{t_1+T}\tan\left(a_1(x(t)-q_1(t)-b_1)\right)dtdt_1 + \\ & + \frac{r_2}{ml}\int_{t_2}^{t_2+T}\int_{t_1}^{t_1+T}\tan\left(a_2(x(t)-q_2(t)-b_2)\right)dtdt_1. \end{aligned} \tag{44}$$

According to the methodology described in Section 3.2, the integrals in (44) are approximated at time $t_n = kT_s$ using the sampled data $q_1(kT_s), q_2(kT_s)$ and $y(kT_s)$, leading to following the predictor for the link's position:

$$\begin{aligned} \hat{y}(kT_s+2T,\boldsymbol{\theta}) = & 2y(kT_s+T) - y(kT_s) + \\ & - c\frac{T_s}{ml}\sum_{k_2=0}^{N_p-1}(y(kT_s+k_2T_s+T) - y(kT_s+k_2T_s)) + \\ & - gT_s^2\sum_{k_2=0}^{N_p-1}\sum_{k_1=0}^{N_p-1}\sin(y(kT_s+k_2T_s+k_1T_s)) + \\ & + r_1\frac{T_s^2}{ml}\sum_{k_2=0}^{N_p-1}\sum_{k_1=0}^{N_p-1}\tan\left(a_1(y(kT_s+k_2T_s+k_1T_s) - q_1(kT_s+k_2T_s+k_1T_s) - b_1)\right) + \\ & + r_2\frac{T_s^2}{ml}\sum_{k_2=0}^{N_p-1}\sum_{k_1=0}^{N_p-1}\tan\left(a_2(y(kT_s+k_2T_s+k_1T_s) - q_2(kT_s+k_2T_s+k_1T_s) - b_2)\right), \end{aligned} \tag{45}$$

Algorithm 1 Hybrid particle swarm optimization and linear regression for estimation of the model's parameters $\boldsymbol{\theta} = [c\ r_1\ r_2\ a_1\ a_2\ b_1\ b_2]$.

Input: number of particles M, maximum number of iterations $k_{\max}$, vector $\boldsymbol{Y}_f$.

1. **populate** particle swarm $(a_1^i, a_2^i, b_1^i, b_2^i)$, $i = 1, \ldots, M$, with random initial values;
2. **for** $k = 1, \ldots, k_{\max}$ **do**
   2.1. **for** $i = 1, \ldots, M$ **do**
      2.1.1. **construct regressors** $\boldsymbol{\phi}^i(kT_s + 2T)$ as in (49), $k = 0, \ldots, \tilde{N}$;
      2.1.2. **compute** the parameters c^i, r_1^i, r_2^i as in (48);
      2.1.3. **set** the parameter vector $\boldsymbol{\theta}^i = \left[c^i\ r_1^i\ r_2^i\ a_1^i, a_2^i\ b_1^i\ b_2^i\right]$;
      2.1.4. **compute** the loss function

$$J(\boldsymbol{\theta}^i) = \sum_{k=0}^{\tilde{N}} \left(y(kT_s + 2T) - \hat{y}(kT_s + 2T, \boldsymbol{\theta}^i)\right)^2;$$

   2.2. **end for**;
   2.3. **choose** the best parameter vector $\boldsymbol{\theta}^* = \arg \min_{\{\boldsymbol{\theta}^i\}_{i=1}^M} J(\boldsymbol{\theta}^i)$;
   2.4. **extract** position $(a_1^*, a_2^*, b_1^*, b_2^*)$ of the best particle from $\boldsymbol{\theta}^*$;
   2.5. **update** particles' position as in (Poli, Kennedy, & Blackwell, 2007, Algorithm 1);
3. **end for**;
4. **end**.

Output: Best particle $\boldsymbol{\theta}^*$.

with $\boldsymbol{\theta} = [c\ r_1\ r_2\ a_1\ a_2\ b_1\ b_2]$. By defining the loss function

$$J(\boldsymbol{\theta}) = \sum_{k=0}^{\tilde{N}} \left(y(kT_s + 2T) - \hat{y}(kT_s + 2T, \boldsymbol{\theta})\right)^2, \tag{46}$$

the least-squares estimator is thus given by the solution of the optimization problem

$$\hat{\boldsymbol{\theta}}_{\text{LS}} = \min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}). \tag{47}$$

Because of the nonlinear dependence of the predictor $\hat{y}(kT_s+2T)$ on the parameters a_1, a_2, b_1 and b_2, the solution of problem (47) cannot be computed analytically. However, the mixed approach based on linear regression and particle swarm optimization outlined in Algorithm 1 can be used to minimize the cost $J(\boldsymbol{\theta})$.

The main idea of Algorithm 1 is to optimize the parameters a_1, a_2, b_1, b_2 through PSO, while the remaining unknown parameters c, r_1 and r_2 are optimized analytically through linear regression. Indeed, given a particle $(a_1^i, a_2^i, b_1^i, b_2^i)$ (namely, for fixed parameters $a_1 = a_1^i, a_2 = a_2^i, b_1 = b_1^i, b_2 = b_2^i$), the predictor $\hat{y}(kT_s + 2T, \boldsymbol{\theta})$ in (45) is a linear function of the remaining parameters c, r_1 and r_2, and their optimal values can be thus computed analytically (Algorithm 1, Step 2.1.2) as

$$[c^i\ \ r_1^i\ \ r_2^i]^T = \left(\boldsymbol{\Phi}^{i^T}\boldsymbol{\Phi}^i\right)^{-1}\boldsymbol{\Phi}^{i^T}\boldsymbol{Y}_f, \tag{48}$$

where $\boldsymbol{\Phi}^i$ is the regressor matrix associated to the i-th particle $(a_1^i, a_2^i, b_1^i, b_2^i)$, which is constructed

(Algorithm 1, Step 2.1.1) by stacking the following regressor in its rows:

$$\boldsymbol{\phi}^i(kT_s+2T) = \begin{bmatrix} -\frac{T_s}{ml}\sum_{k_2=0}^{N_p-1}(y(kT_s+k_2T_s+T)-y(kT_s+k_2T_s)) \\ \frac{T_s^2}{ml}\sum_{k_2=0}^{N_p-1}\sum_{k_1=0}^{N_p-1}\tan\left(a_1^i(y(kT_s+k_2T_s+k_1T_s)-q_1(kT_s+k_2T_s+k_1T_s)-b_1^i)\right) \\ \frac{T_s^2}{ml}\sum_{k_2=0}^{N_p-1}\sum_{k_1=0}^{N_p-1}\tan\left(a_2^i(y(kT_s+k_2T_s+k_1T_s)-q_2(kT_s+k_2T_s+k_1T_s)-b_2^i)\right) \end{bmatrix}, \tag{49}$$

and where the rows of the vector $\boldsymbol{Y}_f$ are given by the known term

$$y_f(kT_s+2T) = y(kT_s+2T)-2y(kT_s+T)+y(kT_s)+gT_s^2\sum_{k_2=0}^{N_p-1}\sum_{k_1=0}^{N_p-1}\sin(y(kT_s+k_2T_s+k_1T_s)). \tag{50}$$

The parameters $c^i, r_1^i, r_2^i, a_1^i, a_2^i, b_1^i, b_2^i$ are then stacked in the vector $\boldsymbol{\theta}^i$ and the corresponding loss function $J(\boldsymbol{\theta}^i)$ is computed (Algorithm 1, Steps 2.1.3 and 2.1.4). The global best parameter $\boldsymbol{\theta}^*$ providing the minimum value of the cost $J(\boldsymbol{\theta}^i)$, for $i = 1, \ldots, M$, is selected (Algorithm 1, Step 2.3), from which the global best particle $(a_1^*, a_2^*, b_1^*, b_2^*)$ is extracted (Algorithm 1, Step 2.4). Finally, the positions of all particles are updated based on common rules in PSO (Algorithm 1, Step 2.5). The algorithm is iterated until the maximum number of iterations $k_{\max}$ is reached.

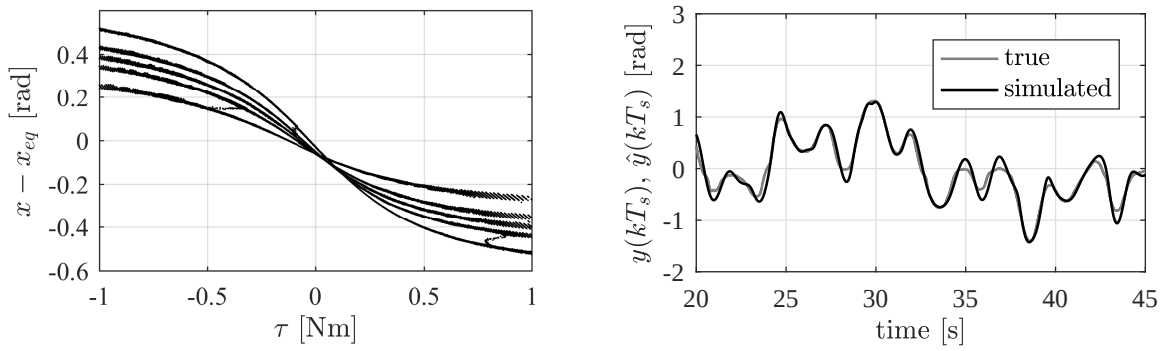
5.3 *Obtained results*

The hybrid linear regression and PSO-based identification algorithm described in Section 5.2 is implemented using $M = 50$ particles and setting the number of iterations $k_{\max} = 5$. Different values of the prediction horizon N_p are tested, and $N_p = 9$ is selected. This value of N_p is the one that maximizes the BFR index on the training dataset. Figure 7(a) shows the estimated torque-displacement characteristic of the VSA (namely, the relation between τ in (43) and the displacement w.r.t. the equilibrium position x_{eq} in (42)). Five different tanh-type functions can be clearly noticed, each one associated to a different stiffness of the actuator.

In order to validate the estimated characteristic of the actuator, the link's position is simulated using the estimated model, and compared with the actual link's position. The obtained results are shown in Figure 7(b). For a better visualization, only a subsequence of validation data is plotted. The achieved BFR on the entire validation dataset is 0.78. For comparison, the case $N_p = 1$ is also considered. This corresponds to the identification of the discretized ODE (43). For $N_p = 1$, the estimated parameters leads to an unstable model with unbounded simulated output. This further shows the advantages of numerical integration w.r.t. the discretization of the ODE (43b).

6. Conclusions

In this paper, we have presented an approach for identification of continuous-time systems based on finite-horizon integration and least-squares estimation. The main idea of the method is to first integrate the *ordinary differential equation* (ODE) describing the system's dynamics over a finite-time window of length T, and then estimate the coefficients of the ODE through least-squares. Although the least-squares estimate is not consistent, the integration horizon T acts as a



(a) Torque-displacement characteristic

(b) Link's position

Figure 7. Identification of VSA. Left panel: estimated torque-displacement characteristic (43a). Right panel: actual link's position $y(kT_s)$ (gray line) and simulated link's position $\hat{y}(kT_s)$ (black line) on validation data.

knob to attenuate the asymptotic bias due to the noise corrupting the output signal observations. A straightforward extension to the identification of nonlinear systems is applied to a real case study concerning the estimation of the nonlinear torque-displacement characteristic of a commercial variable stiffness actuator driving a rotary pendulum. Since the ODE describing the system's behaviour depends nonlinearly on some unknown parameters, a combination of linear regression and particle swarm optimization is used to minimize a least-squares criterion.

The numerical example and the experimental case study reported in the paper have shown the advantages of the presented approach with respect to the identification of a discrete-time model. For instance, in the experimental case study, the identification of the discretized model leads to an unstable model which is not able to reproduce the correct behaviour of the real system.

Current research activities are devoted to: (*i*) extension to the identification of gray-box state-space models; (*ii*) quantification (and eventually correction) of the bias in presence of nonlinear distortions of the noise; (*iii*) application to parameter estimation in partial differential equations from data sampled in time and space.

Acknowledgements

The author is grateful to *qb-robotics s.r.l.* for providing the laboratory equipments used in the experimental case study.

Funding

The work in this paper was partially supported by the EU H2020 project "DAEDALUS - Distributed control and simulation platform to support an ecosystem of digital automation developers", grant no. 723248.

Supplemental material

Datasets and MATLAB codes used in the examples are available for download at `http://dariopiga.com/Software/CTid/IJC/Matlab.rar`

References

Bastogne, T., Garnier, H., & Sibille, P. (2001). A PMF-based subspace method for continuous-time model identification. application to a multivariable winding process. *International Journal of Control*, *74*(2), 118–132.

Catalano, M. G., Grioli, G., Garabini, M., Bonomo, F., Mancini, M., Tsagarakis, N., & Bicchi, A. (2011). VSA-cubebot: a modular variable stiffness platform for multiple degrees of freedom robots. In *IEEE International Conference on Robotics and Automation (ICRA)* (pp. 5090–5095).

Daniel-Berhe, S., & Unbehauen, H. (1998). Bilinear continuous-time systems identification via hartley-based modulating functions. *Automatica*, *34*(4), 499–503.

Della Santina, C., Piazza, C., Gasparri, G. M., Bonilla, M., Catalano, M. G., Grioli, G., ... Bicchi, A. (2017). The quest for natural machine motion: An open platform to fast-prototyping articulated soft robots. *IEEE Robotics & Automation Magazine*, *24*(1), 48–56.

Garnier, H. (2015). Direct continuous-time approaches to system identification. overview and benefits for practical applications. *European Journal of control*, *24*, 50–62.

Garnier, H., Gilson, M., Young, P. C., & Huselstein, E. (2007). An optimal IV technique for identifying continuous-time transfer function model of multiple input systems. *Control engineering practice*, *15*(4), 471–486.

Garnier, H., Mensler, M., & Richard, A. (2003). Continuous-time model identification from sampled data: implementation issues and performance evaluation. *International journal of Control*, *76*(13), 1337–1357.

Garnier, H., & Wang, L. (2008). *Identification of continuous-time models from sampled data* (1st ed.). Springer Publishing Company, Incorporated.

Garnier, H., & Young, P. (2014a). The advantages of directly identifying continuous-time transfer function models in practical applications. *International Journal of Control*, *87*(7), 1319–1338.

Garnier, H., & Young, P. C. (2012). What does continuous-time model identification have to offer? *IFAC Proceedings Volumes*, *45*(16), 810–815.

Garnier, H., & Young, P. C. (2014b). *Special issue on applications of continuous-time model identification and estimation.* Taylor & Francis.

Jemni, A., & Trigeassou, J. C. (1996). A comparative study of the deterministic accuracy of two identification methods for continuous systems. In *IMACS Multiconference: computational engineering in systems applications* (pp. 488–493). Lille, France.

Lataire, J., Pintelon, R., Piga, D., & Tóth, R. (2017). Continuous-time linear time-varying system identification with a frequency-domain kernel-based estimator. *IET Control Theory Applications*, *11*(4), 457-465.

Lennart, L., & Singh, R. (2012). Version 8 of the MATLAB system identification toolbox. In *16th IFAC Symposium on System Identification* (pp. 1826–1831). Brussels, Belgium.

Ljung, L. (1999). *System identification, theory for the user.* Upper Saddle River: Prentince Hall.

Mercère, G., Palsson, H., & Poinot, T. (2011). Continuous-time linear parameter-varying identification of a cross flow heat exchanger: a local approach. *IEEE Transactions on Control Systems Technology*, *19*(1), 64–76.

Padilla, A., Garnier, H., & Gilson, M. (2015). Version 7.0 of the CONTSID toolbox. In *17th IFAC Symposium on System Identification.* Beijing, China.

Patra, A., & Unbehauen, H. (1995). Identification of a class of nonlinear continuous-time systems using Hartley modulating functions. *International Journal of Control*, *62*(6), 1431–1451.

Poli, R., Kennedy, J., & Blackwell, T. (2007). Particle swarm optimization. *Swarm intelligence*, *1*(1), 33–57.

Rao, G. P., & Garnier, H. (2002). Numerical illustrations of the relevance of direct continuous-time model identification. In *15th Triennial IFAC World Congress on Automatic Control*

(pp. 133–138).

Rao, G. P., & Unbehauen, H. (2006). Identification of continuous-time systems. *IEE Proceedings-Control theory and applications*, *153*(2), 185–220.

Sagara, S., & Zhao, Z. Y. (1990). Numerical integration approach to on-line identification of continuous-time systems. *Automatica*, *26*(1), 63–74.

Shanghong, H., & Jue, Z. (2004). Identification of linear continuous-time system using wavelet modulating filters. *Journal of Systems Engineering and Electronics*, *15*(3), 270–277.

Sinha, N. K., & Rao, G. P. (1991). *Identification of continuous-time systems. methodology and computer implementation* (Vol. 161). Kluwer Academic Publishers.

Söderström, T., & Stoica, P. (1983). *Instrumental variable methods for system identification* (Vol. 161). Springer-Verlag Berlin.

Young, P., & Jakeman, A. (1980). Refined instrumental variable methods of recursive time-series analysis part iii. extensions. *International Journal of Control*, *31*(4), 741–764.

Young, P. C., & Garnier, H. (2006). Identification and estimation of continuous-time, data-based mechanistic (DBM) models for environmental systems. *Environmental modelling & software*, *21*(8), 1055–1072.