

# Identifying Lebesgue-sampled Continuous-time Impulse Response Models: A Kernel-based Approach<sup>★</sup>

Rodrigo A. González<sup>\*</sup> Koen Tiels<sup>\*</sup> Tom Oomen<sup>\*,\*\*</sup>

<sup>\*</sup> *Control Systems Technology Section, Department of Mechanical Engineering, Eindhoven University of Technology, The Netherlands.  
(e-mails: r.a.gonzalez@tue.nl; k.tiels@tue.nl; t.a.e.oomen@tue.nl).*

<sup>\*\*</sup> *Delft Center for Systems and Control, Delft University of Technology, The Netherlands*

---

**Abstract:** Control applications are increasingly sampled non-equidistantly in time, including in motion control, networked control, resource-aware control, and event-triggered control. Some of these applications use measurement devices that sample equidistantly in the amplitude domain. The aim of this paper is to develop a non-parametric estimator of the impulse response of continuous-time systems based on such sampling strategy, known as Lebesgue-sampling. To this end, kernel methods are developed to formulate an algorithm that adequately takes into account the output intersample behavior, which ultimately leads to more accurate models and more efficient output sampling compared to the standard approach. The efficacy of this method is demonstrated through a mass-spring damper case study.

*Keywords:* System identification; continuous-time systems; event-based sampling.

---

## 1. INTRODUCTION

In system identification and control, it is common to assume that the signals are sampled equidistantly in time, also known as Riemann sampling. In contrast, event-based sampling can lead to improvements in control performance, as well as in resource efficiency (Åström and Bernhardsson, 2003). In particular, Lebesgue sampling is one of the most popular event-based sampling methods, and it consists of sampling a signal whenever it crosses fixed thresholds in the amplitude domain. This sampling can be found in incremental encoders (Merry et al., 2013), and also in networked control systems, where the goal is to reduce resource utilization without affecting throughput.

The Lebesgue sampling paradigm provides knowledge on what amplitude band the signals are located in at each instant of time, thus yielding a quantized measurement whenever the signal does not cross a threshold. A considerable effort has been devoted to the identification of systems based on quantized measurements. The maximum likelihood estimator based on the expectation-maximization algorithm (EM) has been derived for finite-impulse response (FIR) systems by Godoy et al. (2011), while Chen et al. (2012) developed a regularized FIR estimator for binary measurements. Risuleo et al. (2019) studied an approximate maximum likelihood approach, and Bottegal et al. (2017) propose a kernel-based method for estimating FIR models. Other approaches have been pursued for the identification of ARX systems (Agüero et al., 2017).

Although there has been recent work on non-parametric identification for continuous-time systems using kernel methods that use non-equidistantly sampled data (Scandella et al., 2022), these works do not incorporate the intersample behavior information derived from a Lebesgue sampling framework (i.e., the lower and upper bounds on the unsampled output in between the time-stamps), which leads to performance detriment. Other work (Kawaguchi et al., 2016) has considered continuous-time systems with Lebesgue-sampled outputs, although such results are only valid for parametric models. On the other hand, the approaches by Chen et al. (2012); Bottegal et al. (2017); Risuleo et al. (2019) for identification with quantized data might be used for obtaining a non-parametric discrete-time model that can later be converted into continuous-time. However, this conversion is in many cases ill-defined and does not admit an arbitrary input intersample behavior, which drives the need for directly estimating a continuous-time system from the data (Garnier and Young, 2014).

The problem that is addressed in this paper is the estimation of non-parametric continuous-time models from Lebesgue-sampled output data. To this end, we seek estimators that can 1) provide a *continuous-time* impulse response estimate from possibly noisy and short data records, and 2) exploit the entirety of the output information contained in the irregular sampling instants and the set-constrained intersample behavior. In summary, the main contributions of this paper are:

- (C1) We propose a non-parametric continuous-time impulse response estimator for Lebesgue-sampled systems. This estimator is obtained by first introducing a loss function that incorporates the output intersam-

---

<sup>★</sup> This work is part of the research program VIDI with project number 15698, which is (partly) financed by The Netherlands Organization for Scientific Research (NWO).

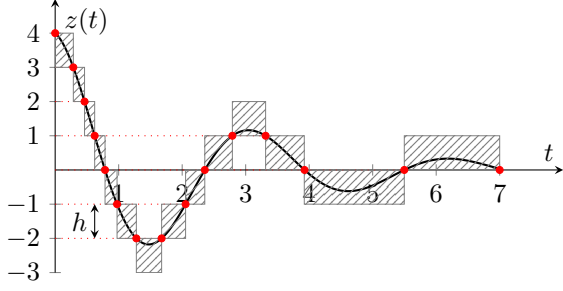


Fig. 1. Lebesgue sampling of a signal  $z(t)$  with threshold  $h = 1$ . Red dots indicate the sampling instants and thresholds being crossed, and dashed gray rectangles show the regions where  $z(t)$  is known to be located.

ple information together with a regularization term. The optimum of the infinite-dimensional optimization problem is characterized by the representer theorem (Schölkopf et al., 2001).

- (C2) Once the kernel-regularized estimator is written as a finite linear combination of representer, we present an iterative procedure that delivers the associated weights based on the *maximum a posteriori* EM (MAP-EM) method. This is done by relating the optimization problem for computing the weights to the MAP estimator of a particular FIR model.
- (C3) We introduce an iterative scheme to optimize the hyperparameters that describe the kernel and noise variance from the Empirical Bayes (EB) approach. These iterations also stem from MAP-EM and require the computation of second-order moments of the unsampled output, which are approximated via a minimax tilting algorithm (Botev, 2017).

The remainder of the paper is organized as follows: in Section 2 the problem of interest is stated, and practical aspects of Lebesgue-sampled system identification are covered. Section 3 contains the derivation of a kernel-based estimator for continuous-time Lebesgue-sampled system identification. A numerical study is presented in Section 4, while Section 5 provides concluding remarks. Proofs of several theoretical results can be found in the Appendix.

## 2. PROBLEM FORMULATION

Consider the following linear and time-invariant (LTI), stable, strictly causal, continuous-time system

$$x(t) = \int_0^\infty g(\tau)u(t-\tau)d\tau, \quad (1)$$

where  $u$  is a causal, deterministic and exogenous input, and  $g$  is the impulse response of the LTI system. The output  $x(t)$  is corrupted by additive measurement noise, which results in a continuous-time signal  $z(t)$ . Assume that we have access to  $N_L$  data points of the Lebesgue-sampled version of  $z(t)$ , as in Fig. 1. That is, given the threshold distance  $h > 0$  and the signal  $z(t)$ , we have at our disposal the sampled sequence  $\{y_L(t_l)\}_{l=1}^{N_L}$  that satisfies  $y_L(t_l) = z(t_l)$ . The time-stamps  $t_l, l = 1, 2, \dots, N_L$ , are the instants in time at which  $z(t)$  crosses a fixed threshold  $hm_l$ , with  $m_l \in \mathbb{Z}$ . Formally, we characterize the time-stamps by

$$t_l = \min \{ \tau \in (t_{l-1}, \infty) : z(\tau) = mh \text{ for some } m \in \mathbb{Z} \},$$

with  $m_l = z(t_l)/h$ . Without loss of generality, we assume that  $t_1 = 0$ . We pursue a non-parametric estimate of  $g$

using the continuous-time input  $\{u(t)\}_{t \in [0, t_{N_L}]}$  and the Lebesgue-sampled output  $\{y_L(t_l)\}_{l=1}^{N_L}$ .

### 2.1 Practical framework for Lebesgue-sampled systems

Incremental encoders operate on this kind of sampling principle (Merry et al., 2013). In practice, a light source emits a beam directed towards a slotted disk or strip, and the output of two light detectors are recorded. These two signals allow the encoder to detect the direction of the rotation. The quantity  $h$  represents the uncertainty in the measurements of the incremental encoder, which is inversely proportional to its resolution. In low-resolution incremental encoders, the quantization effect produced by  $h$ , in conjunction with the non-equidistant nature of the sampling mechanism, can impact the performance of, e.g., iterative learning control (Strijbosch and Oomen, 2022).

Behind this sampling procedure there typically is an amplitude detection mechanism that operates at a fast sampling rate. With this context in mind, we define  $\Delta > 0$  as the sampling period of this amplitude detection mechanism. The following assumption is set in place.

*Assumption 1.* For every time instant  $t = i\Delta$ ,  $i = 0, 1, \dots, \lfloor t_{N_L}/\Delta \rfloor + 1$ , the lower and upper threshold levels associated with the unsampled output  $z(t)$  are known. The lower bound at each time-instant  $t = i\Delta$  is denoted as  $\eta_i$ , and it can be deduced unambiguously from  $\{y_L(t_l)\}_{l=1}^{N_L}$ .

Thus, a *set-valued* signal that represents a fast-sampled version of the gray rectangles in Fig. 1 can be defined as

$$y(i\Delta) = \mathcal{Q}_h\{z(i\Delta)\} := [\eta_i, \eta_i + h]. \quad (2)$$

In the proposed setup, the practitioner has access to  $y(i\Delta)$  as output information. To simplify our exposition (and with some abuse of notation), we denote  $\{y(i\Delta)\}_{i=0}^N$  and  $\{z(i\Delta)\}_{i=0}^N$  as  $\mathbf{y}_{0:N}$  and  $\mathbf{z}_{0:N}$ , respectively, with  $N := \lfloor t_{N_L}/\Delta \rfloor + 1$ . Note that all vectors and matrices are written in bold throughout the paper.

Note that we do not assume that each time-stamp  $t_l$  is a multiple of  $\Delta$ . Although such assumption is commonly used in intermittent sampling setups (Strijbosch and Oomen, 2022), we do not require it in this paper.

*Assumption 2.* At every time instant  $t = i\Delta$ , the disturbance noise affecting the output  $z(i\Delta)$  is an additive discrete-time independent and identically distributed (i.i.d.) Gaussian white noise of variance  $\sigma^2$ .

The noise variance is not known beforehand and is typically estimated from the data or selected *a priori* according to expert knowledge. After taking into consideration Assumptions 1 and 2, the problem of interest is: Considering the continuous-time input  $\{u(t)\}_{t \in [0, t_{N_L}]}$  and the set-valued data  $\{y(i\Delta)\}_{i=0}^N$ , obtain a non-parametric estimate of the underlying continuous-time impulse response  $g$ .

*Remark 1.* In many cases, the input  $u$  is generated by a zero-order-hold device of sampling period  $\Delta_u$ . When  $\Delta_u$  is a multiple of  $\Delta$ , we may consider the sampled input signal  $\{u(i\Delta)\}_{i=0}^N$  instead of a fully continuous-time description. Both viewpoints are equivalent if the intersample behavior of the sampled input is known and correctly incorporated in the construction of the algorithms.

*Remark 2.* We will only consider the output data that are produced by the input excitation starting from  $t = 0$ . We discard the first output measurement  $y(0)$  due to causality.

### 3. NON-PARAMETRIC ESTIMATION USING LEBESGUE-SAMPLED DATA

This paper solves the identification problem by incorporating the Lebesgue-sampling strategy into the framework of continuous-time kernel-based methods. This section addresses the main challenges regarding how to adequately exploit the output uncertainty set information instead of point data, and how to tune the kernel hyperparameters.

#### 3.1 MAP estimator for Lebesgue-sampled systems

We begin by addressing the Lebesgue-sampled system identification problem from a Bayesian standpoint. In this regard, we assume that  $g$  can be modeled by a zero-mean Gaussian process with prior distribution  $p(g)$  and covariance  $\mathbb{E}\{g(t)g(s)\} = \gamma^{-1}k(t, s)$  with  $\gamma > 0$ , and we are interested in computing the MAP-inspired estimator

$$\hat{g}_{\text{MAP}}(t) = \arg \max_g (\ell(g) + \log p(g)), \quad (3)$$

where  $\ell(\cdot)$  denotes the log-likelihood function  $\ell(g) = \log p(\mathbf{y}_{1:N}|g)$ , with  $p$  denoting the probability density function. The prior  $p(g)$  must be described in some manner so that (3) is tractable. Although the prior density cannot be formally defined in infinite-dimensional function spaces, there exists a formal connection (Aravkin et al., 2014) between MAP estimation and kernel methods by setting the prior density being proportional to  $\exp(-\gamma\|g\|_{\mathcal{G}}^2)$ , where  $\mathcal{G}$  is a reproducing kernel Hilbert space (RKHS) that has  $k$  as its kernel. Thus, the first result we present concerns finding transparent expressions for

$$\hat{g} = \arg \min_{g \in \mathcal{G}} (-\log p(\mathbf{y}_{1:N}|g) + \gamma\|g\|_{\mathcal{G}}^2), \quad (4)$$

for a *fixed* kernel  $k$ , when  $\mathbf{y}_{1:N}$  is Lebesgue-sampled data. The next theorem constitutes Contribution C1 of this work.

*Theorem 3.* Under Assumptions 1 and 2, the kernel-based estimator (4) is given by

$$\hat{g}(t) = \sum_{i=1}^N c_i \hat{g}_i(t), \quad (5)$$

where

$$\hat{g}_i(t) = \int_0^\infty u(i\Delta - \tau)k(t, \tau)d\tau, \quad (6)$$

and the vector of coefficients  $\hat{\mathbf{c}} := [c_1, c_2, \dots, c_N]^\top$  is obtained by solving the optimization problem

$$\hat{\mathbf{c}} = \arg \min_{\mathbf{c} \in \mathbb{R}^N} \left( -\sum_{i=1}^N \log \left[ \int_{\eta_i}^{\eta_i+h} e^{-\frac{1}{2\sigma^2}(z_i - \mathbf{K}_i^\top \mathbf{c})^2} dz_i \right] + \gamma \mathbf{c}^\top \mathbf{K} \mathbf{c} \right), \quad (7)$$

with  $\mathbf{K}_i$  being the  $i$ th column of the matrix  $\mathbf{K}$  with entries

$$\mathbf{K}_{ij} = \int_0^\infty \int_0^\infty u(i\Delta - \xi)u(j\Delta - \tau)k(\xi, \tau)d\tau d\xi. \quad (8)$$

**Proof.** We first obtain an expression for  $p(\mathbf{y}_{1:N}|g)$ . The probability density function of the output prior to sampling  $\mathbf{z}_{1:N} = [z(\Delta), \dots, z(N\Delta)]^\top$  (conditioned to  $g$ ) is

$$p(\mathbf{z}_{1:N}|g) = \frac{1}{(2\pi\sigma^2)^{\frac{N}{2}}} \prod_{i=1}^N e^{-\frac{1}{2\sigma^2} [z(i\Delta) - (g*u)(i\Delta)]^2},$$

where we have used the fact that the additive noise is Gaussian and i.i.d. by Assumption 2. Therefore, from (2), the probability mass function of  $\mathbf{y}_{1:N}$  is

$$p(\mathbf{y}_{1:N}|g) = \mathbb{P}(z(\Delta) \in [\eta_1, \eta_1 + h), \dots, z(N\Delta) \in [\eta_N, \eta_N + h)|g) \\ = \frac{1}{(2\pi\sigma^2)^{\frac{N}{2}}} \prod_{i=1}^N \int_{\eta_i}^{\eta_i+h} e^{-\frac{1}{2\sigma^2} [z_i - (g*u)(i\Delta)]^2} dz_i. \quad (9)$$

Therefore, computing (4) is equivalent to solving

$$\min_{g \in \mathcal{G}} \left( -\sum_{i=1}^N \log \left[ \int_{\eta_i}^{\eta_i+h} e^{-\frac{1}{2\sigma^2} [z_i - (g*u)(i\Delta)]^2} dz_i \right] + \gamma\|g\|_{\mathcal{G}}^2 \right). \quad (10)$$

By the representer theorem (Schölkopf et al., 2001), any  $g \in \mathcal{G}$  that minimizes the cost in (10) admits a representation of the form (5) with representers  $\hat{g}_i$  in (6). Note that any minimizer  $g$  satisfies  $(g*u)(i\Delta) = \mathbf{K}_i^\top \mathbf{c}$ , with  $\mathbf{K}_i$  being the  $i$ th column of the kernel matrix described by (8), and also  $\|g\|_{\mathcal{G}}^2 = \mathbf{c}^\top \mathbf{K} \mathbf{c}$ . Thus, replacing the finite-dimensional representation (5) in (10) reduces the optimization problem to finding  $\mathbf{c}$  that minimizes (7).  $\square$

*Remark 3.* Although  $\hat{g}$  in (5) is not strictly speaking a MAP estimator of the impulse response  $g$ , it can be shown that  $(\hat{g}*u)$  provides a MAP estimator of the output prior to Lebesgue sampling at the instants  $t = i\Delta$ ,  $i = 1, \dots, N$ . The proof of this result, which complements the one in Aravkin et al. (2014) for Riemann-sampled data, is out of the scope of this paper and will be published elsewhere.

The main contribution of Theorem 3 is recasting the infinite-dimensional problem for Lebesgue-sampled system identification in (4) as a finite dimensional one in (7). Note that  $\hat{\mathbf{c}}$  does not have an explicit form as the Riemann-sampling counterpart (Pillonetto et al., 2022), where it is computed directly as the solution of a regularized least-squares problem. The following subsection is focused on how to compute  $\hat{\mathbf{c}}$  for a fixed kernel  $k$  and hyperparameters  $\gamma$  and  $\sigma^2$ . Later, we present an algorithm for computing the kernel hyperparameters using Empirical Bayes.

#### 3.2 Optimal weights with MAP-EM

To compute (7), we propose an iterative procedure based on the MAP-EM algorithm. This procedure, which ensures the computation of a local maximum of the cost in (7) under general conditions as an extension of the standard EM (Wu, 1983), constitutes Contribution C2 of this paper. The approach exploits the fact that (7) is related to the MAP estimator of an FIR model, which is used for applying the EM algorithm tailored for MAP estimation. The following lemma makes this connection evident.

*Lemma 4.* Consider the following linear regression model with a set-valued output:

$$z(i\Delta) = \mathbf{K}_i^\top \mathbf{c} + e(i\Delta), \quad (11a)$$

$$y(i\Delta) = \mathcal{Q}_h\{z(i\Delta)\}, \quad (11b)$$

where  $\{e(i\Delta)\}_{i=1}^N$  is an i.i.d. Gaussian white noise with variance  $\sigma^2$ , and  $\mathbf{K}_i$ ,  $i = 1, 2, \dots, N$ , is assumed known. Assume that  $\mathbf{c}$  in (11a) has a Gaussian prior distribution, with zero mean and covariance  $\mathbf{K}^{-1}/(2\gamma)$ . Then, the MAP estimator for  $\mathbf{c}$  is given by  $\hat{\mathbf{c}}$  in (7).

**Proof.** See Appendix A.  $\square$

By Lemma 4 we can view the computation of the weights  $\hat{\mathbf{c}}$  in a MAP-EM framework if we set the unquantized output  $z(i\Delta)$  in (11a) as our hidden variable. In other words, we can optimize the *a posteriori* density for  $\mathbf{c}$  (which is exactly the objective function in (7)) by computing the conditional expectation (i.e., the E-step) of the log complete-data posterior density given the measurements  $\mathbf{y}_{1:N}$  and the current estimate of  $\hat{\mathbf{c}}$ , and later performing a maximization step (i.e., the M-step). These two steps are outlined in Algorithm 1. Details pertaining the EM algorithm and its derivation can be found in, e.g., (Dempster et al., 1977).

---

**Algorithm 1** MAP-EM for computing  $\hat{\mathbf{c}}$  in (7)

---

- 1: Input: initial estimate  $\hat{\mathbf{c}}^{(1)}$
  - 2: **for**  $j = 1, 2, \dots$  **do**
  - 3:   **E-step:** Compute the expectation
 
$$Q(\mathbf{c}, \hat{\mathbf{c}}^{(j)}) = \mathbb{E}\{\log p(\mathbf{z}_{1:N}, \mathbf{y}_{1:N} | \mathbf{c}) | \mathbf{y}_{1:N}, \hat{\mathbf{c}}^{(j)}\} \quad (12)$$
  - 4:   **M-step:** Solve the optimization problem
 
$$\hat{\mathbf{c}}^{(j+1)} = \arg \max_{\mathbf{c} \in \mathbb{R}^N} \left( Q(\mathbf{c}, \hat{\mathbf{c}}^{(j)}) - \gamma \mathbf{c}^\top \mathbf{K} \mathbf{c} \right) \quad (13)$$
  - 5: **end for**
- 

The E-step is obtained from a result from quantized FIR estimation. Afterwards, we present the M-step in Theorem 6, which contains the iterations for computing  $\hat{\mathbf{c}}$ .

*Lemma 5.* (Godoy et al., 2011, Lemma 5). Consider the discrete-time model (11). The  $Q$  function in (12) satisfies

$$Q(\mathbf{c}, \hat{\mathbf{c}}^{(j)}) = \frac{-1}{2\sigma^2} \sum_{i=1}^N \int_{\eta_i}^{\eta_i+h} (z_i - \mathbf{K}_i^\top \mathbf{c})^2 p(z_i | y(i\Delta), \hat{\mathbf{c}}^{(j)}) dz_i + C,$$

where  $C$  is a constant.

**Proof.** See Godoy et al. (2011).  $\square$

*Theorem 6.* The M-step in (13) is equivalent to

$$\hat{\mathbf{c}}^{(j+1)} = (\mathbf{K} + \tilde{\gamma} \mathbf{I})^{-1} \tilde{\mathbf{z}}^{(j)}, \quad (14)$$

where  $\tilde{\gamma} = 2\sigma^2\gamma$ , and with the  $i$ th entry of  $\tilde{\mathbf{z}}^{(j)}$  being

$$\tilde{z}_i^{(j)} = \mathbf{K}_i^\top \hat{\mathbf{c}}^{(j)} + \frac{\sqrt{\frac{2}{\pi}} \sigma \left( \exp\{-(b_i^{(j)})^2\} - \exp\{-(b_i^{(j)} + \frac{h}{\sqrt{2}\sigma})^2\} \right)}{\text{erf}[b_i^{(j)} + \frac{h}{\sqrt{2}\sigma}] - \text{erf}[b_i^{(j)}]}, \quad (15)$$

where  $b_i^{(j)} := (\eta_i - \mathbf{K}_i^\top \hat{\mathbf{c}}^{(j)})/(\sqrt{2}\sigma)$ , and the error function is defined by  $\text{erf}[x] := (2/\sqrt{\pi}) \int_0^x \exp(-t^2) dt$ .

**Proof.** See Appendix B.  $\square$

### 3.3 Kernel hyper-parameter optimization

The optimal weighting  $\hat{\mathbf{c}}$  depends on the kernel that is selected (together with the regularization constant  $\gamma$ ), and of the noise covariance  $\sigma^2$ . These unknown quantities are typically encompassed in a hyperparameter vector  $\boldsymbol{\rho}$  that must be estimated using the data at hand. We pursue an Empirical Bayes (EB) approach to estimate these unknown parameters. The EB algorithm is highly efficient, as is evidenced in its broad application (Pillonetto et al., 2014; Scandella et al., 2022).

There are many well-known kernels that are used for continuous-time impulse response estimation. For example, the stable-spline one of order  $q$  is defined as

$$k(t, \tau) = s_q(e^{-\beta t}, e^{-\beta \tau}), \quad q \in \mathbb{N},$$

where  $\beta$  is a positive hyperparameter, and  $s_q$  is the regular spline kernel of order  $q$ . For  $q = 1$  we have  $s_1(e^{-\beta t}, e^{-\beta \tau}) = e^{-\beta \max(t, \tau)}$ . The EB approach involves estimating the vector  $\boldsymbol{\rho} = [\gamma, \beta, \sigma^2]^\top$  by solving the optimization problem

$$\hat{\boldsymbol{\rho}}_{\text{EB}} = \arg \max_{\boldsymbol{\rho} \in \boldsymbol{\Gamma}} p(\mathbf{y}_{1:N} | \boldsymbol{\rho}), \quad (16)$$

where  $\boldsymbol{\Gamma}$  denotes the admissible space of hyperparameters (i.e.,  $\gamma, \beta, \sigma^2 > 0$  in the stable-spline case). To describe such optimization problem more explicitly, we first compute the probability density function of the output prior to Lebesgue sampling,  $p(\mathbf{z}_{1:N} | \boldsymbol{\rho})$ . This expression can be obtained directly by exploiting the fact that the additive noise is assumed Gaussian and independent of  $g$  (which is also assumed Gaussian), which leads to

$$\mathbf{z}_{1:N} | \boldsymbol{\rho} \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_\beta / \gamma + \sigma^2 \mathbf{I}), \quad (17)$$

where we have made explicit the dependence of the kernel matrix  $\mathbf{K}$  on the hyperparameter  $\beta$ . Therefore, the EB estimator for  $\boldsymbol{\rho}$  is given by

$$\hat{\boldsymbol{\rho}}_{\text{EB}} = \arg \max_{\boldsymbol{\rho} \in \boldsymbol{\Gamma}} \frac{1}{\sqrt{\det(2\pi[\mathbf{K}_\beta / \gamma + \sigma^2 \mathbf{I}])}} \times \int_{\mathbf{z} \in \mathbf{y}_{1:N}} \exp\left\{-\frac{1}{2} \mathbf{z}^\top (\mathbf{K}_\beta / \gamma + \sigma^2 \mathbf{I})^{-1} \mathbf{z}\right\} d\mathbf{z}, \quad (18)$$

where the elements of  $\mathbf{y}_{1:N}$  are defined in (2). This optimization problem involves an  $N$ -dimensional integral, which is hard to compute in general (see, e.g., Chen et al. (2012); Bottegal et al. (2017)). The intractability is here addressed by optimizing (18) with EM, similarly as in the previous subsection. For brevity, we derive the EM iterations jointly (both E and M steps) in Theorem 7.

*Theorem 7.* The following iterative procedure is guaranteed to converge with probability 1 to a (local or global) maximum for the cost in (18):

$$\hat{\boldsymbol{\rho}}^{(j+1)} = \arg \min_{\boldsymbol{\rho} \in \boldsymbol{\Gamma}} \left( \log \det(\mathbf{S}_\rho) + \text{tr}\{\mathbf{S}_\rho^{-1} \bar{\mathbf{Q}}^{(j)}\} \right), \quad (19)$$

where  $\mathbf{S}_\rho := \mathbf{K}_\beta / \gamma + \sigma^2 \mathbf{I}$ , and  $\bar{\mathbf{Q}}^{(j)}$  is the second moment of  $\mathbf{z}_{1:N}$  given the data and the  $j$ th iteration of  $\hat{\boldsymbol{\rho}}$ , i.e.,

$$\bar{\mathbf{Q}}^{(j)} = \mathbb{E}\{\mathbf{z}_{1:N} \mathbf{z}_{1:N}^\top | \mathbf{y}_{1:N}, \hat{\boldsymbol{\rho}}^{(j)}\}. \quad (20)$$

**Proof.** See Appendix C.  $\square$

The iterations in Theorem 7 reduce the  $N$ -dimensional integral optimization problem into a form similar to the Empirical Bayes optimization problem (Pillonetto et al., 2014) for unquantized output data, at the expense of having to compute the second moment of  $\mathbf{z}_{1:N}$  conditioned on the set-valued data. The  $\bar{\mathbf{Q}}^{(j)}$  matrix cannot be computed in closed-form in general. In this paper, we extract samples of a multivariate truncated Gaussian distribution using the algorithm in Botev (2017) and we approximate the expectation in (20) via Monte Carlo integration.

### 3.4 The complete algorithm

To conclude this section, the full algorithm for non-parametric identification of Lebesgue-sampled continuous-time systems is described in Algorithm 2.

## 4. CASE STUDY

In this section, the novel non-parametric estimator is tested on a practically-relevant example. We consider a

---

**Algorithm 2** Kernel-based non-parametric identification for Lebesgue-sampled continuous-time systems

---

- 1: Input:  $\mathbf{u}_{0:N-1}, \mathbf{y}_{1:N}$ , initial hyperparameter estimate  $\hat{\rho}^{(1)} = [\hat{\gamma}^{(1)}, \hat{\beta}^{(1)}, \hat{\sigma}^{2(1)}]^\top$ , initial weighting estimate  $\hat{\mathbf{c}}^{(1)}$ , maximum number of MAP-EM iterations  $M$
  - 2: **for**  $j = 1, 2, \dots, M$  **do**
  - 3:   Compute  $\mathbf{Q}^{(j)}$  from (20) using Botev (2017)
  - 4:   Obtain  $\hat{\rho}^{(j+1)} = [\hat{\gamma}^{(j+1)}, \hat{\beta}^{(j+1)}, \hat{\sigma}^{2(j+1)}]^\top$  from (19)
  - 5: **end for**
  - 6: **for**  $j = 1, 2, \dots, M$  **do**
  - 7:   Obtain  $\hat{\mathbf{c}}^{(j+1)}$  from (14) with  $\mathbf{K}$ ,  $\tilde{\gamma}$  and  $\tilde{\mathbf{z}}^{(j)}$  computed using  $\hat{\rho}^{(M+1)}$
  - 8: **end for**
  - 9: Output:  $\hat{g}$  obtained from (5), with  $\hat{g}_i$  computed from (6) using  $\hat{\beta}^{(M+1)}$  as the hyperparameter for  $k$ .
- 

mass-spring-damper system with transfer function given by

$$G(s) = \frac{1}{ms^2 + ds + k},$$

with mass  $m=0.05[\text{Kg}]$ , damping coefficient  $d=0.2[\text{Ns/m}]$ , and spring constant  $k=1[\text{N/m}]$ . The output is sensed with period  $\Delta = 0.1[\text{s}]$ , and  $h = 1[\text{m}]$ . The input is a Gaussian white noise sequence passed through a zero-order hold device with period  $\Delta_u = 3[\text{s}]$ . One hundred Monte Carlo runs are performed with a varying input and an additive Gaussian white noise prior to the Lebesgue-sampling with standard deviation  $0.05[\text{m}]$ . Each run has a total time duration of  $30[\text{s}]$  (i.e., 300 data points are sensed prior to Lebesgue-sampling), and on average  $N_L = 69$  output samples are obtained after sampling per run.

Three estimators are tested: the standard kernel-based continuous-time non-parametric estimator (Scandella et al., 2022) using the midpoint estimate  $z(i\Delta) \approx \eta_i + h/2$  as output data ( $\hat{g}_{\text{rie}}$ ), this same estimator but using the noisy output  $z(i\Delta)$  prior to Lebesgue sampling as output data ( $\hat{g}_{\text{or}}$ ), and the proposed approach (Algorithm 2 of this paper,  $\hat{g}_{\text{leb}}$ ). Note that the oracle estimator  $\hat{g}_{\text{or}}$  cannot be implemented in practice, since we do not have direct knowledge of the system output before the event-sampler. We measure each estimator's performance with the metric

$$\text{fit} = 100(1 - \|\hat{\mathbf{x}}^j - \mathbf{x}\|_2 / \|\mathbf{x} - \bar{\mathbf{x}}\mathbf{1}\|_2),$$

where  $\mathbf{x}$  is the noiseless output sequence (prior to Lebesgue-sampling),  $\hat{\mathbf{x}}^j$  is the simulated output using the  $j$ th impulse response estimate, and  $\bar{\mathbf{x}}$  is the mean value of  $\mathbf{x}$ . The proposed estimator uses the 1st order stable-spline kernel with a maximum number of MAP-EM iterations  $M = 40$ , and 1000 samples of a multivariate truncated Gaussian distribution are obtained to compute  $\mathbf{Q}^{(j)}$  in (20).

In Figure 2, a typical data set is shown. Note that the task of the proposed estimator is particularly challenging, since the overshoot of the output signal  $z$  is rarely captured in the  $y$  signal band due to the coarse grid produced by the threshold level. Figure 3 shows the boxplots of the fit metric for each estimator. The proposed approach achieves on average a better fit than the estimator that only uses the midpoint values  $\eta_i + h/2$  as output. The  $\hat{g}_{\text{leb}}$  estimator is only slightly outperformed by the oracle, despite having a low resolution for the output measurement mechanism and a 77% reduction in output data samples on average.

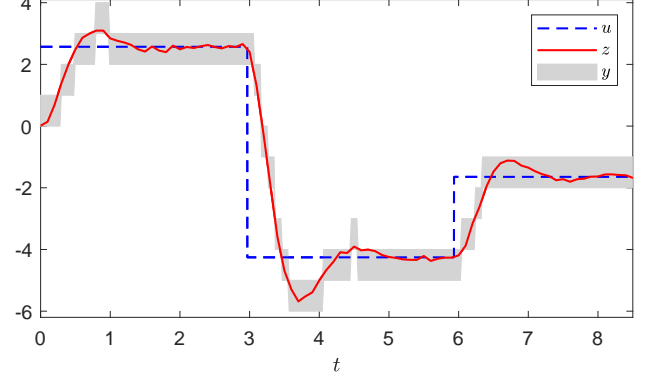


Fig. 2. Input and output signals corresponding to 8[s] of one Monte Carlo run, with  $h = 1$ . The proposed estimator  $\hat{g}_{\text{leb}}$  only uses the fact that the unsampled output is located in the grey band  $y$ .

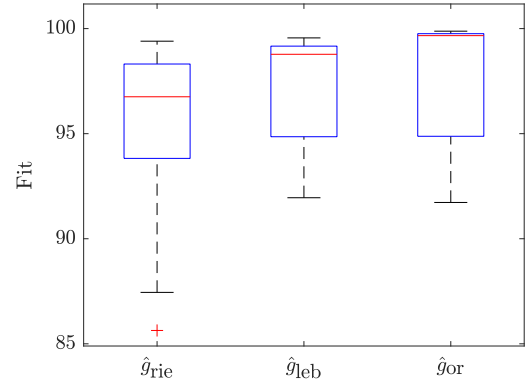


Fig. 3. Boxplots of the fit metric for the case study. The proposed estimator  $\hat{g}_{\text{leb}}$  outperforms the midpoint method  $\hat{g}_{\text{rie}}$ , and is close in performance to the oracle estimator  $\hat{g}_{\text{or}}$ , despite using fewer effective samples.

## 5. CONCLUSIONS

In this paper we have presented a non-parametric method to estimate the impulse response of a continuous-time system based on Lebesgue-sampled data. The proposed algorithm, which is inspired by MAP estimation and kernel methods, exploits the intersample knowledge of the output to deliver more accurate models than the standard approach while needing much fewer output samples. We have showed the efficacy of the method in a practically-relevant example. Future research in this setup includes the study of computationally-efficient algorithms and the allocation of the threshold levels for experiment design.

## REFERENCES

- Agüero, J.C., González, K., and Carvajal, R. (2017). EM-based identification of ARX systems having quantized output data. *IFAC-PapersOnLine*, 50(1), 8367–8372.
- Aravkin, A., Bell, B., Burke, J., and Pilonetto, G. (2014). The connection between Bayesian estimation of a Gaussian random field and RKHS. *IEEE Trans. on Neural Networks and Learning Systems*, 26(7), 1518–1524.
- Åström, K.J. and Bernhardsson, B. (2003). Systems with Lebesgue sampling. In *Directions in Mathematical Systems Theory and Optimization*, 1–13. Springer.
- Botev, Z.I. (2017). The normal law under linear restrictions: simulation and estimation via minimax tilting.

*Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(1), 125–148.

Bottegal, G., Hjalmarsson, H., and Pillonetto, G. (2017). A new kernel-based approach to system identification with quantized output data. *Automatica*, 85, 145–152.

Chen, T., Zhao, Y., and Ljung, L. (2012). Impulse response estimation with binary measurements: A regularized FIR model approach. *IFAC Proceedings Volumes*, 45(16), 113–118.

Dempster, A.P., Laird, N.M., and Rubin, D.B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1), 1–22.

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

Godoy, B.I., Goodwin, G.C., Agüero, J.C., Marelli, D., and Wigren, T. (2011). On identification of FIR systems having quantized output data. *Automatica*, 47(9), 1905–1915.

Kawaguchi, T., Hikono, S., Maruta, I., and Adachi, S. (2016). System identification under Lebesgue sampling and its asymptotic property. In *55th IEEE Conference on Decision and Control*, 2079–2084.

Merry, R.J.E., Van de Molengraft, M.J.G., and Steinbuch, M. (2013). Optimal higher-order encoder time-stamping. *Mechatronics*, 23(5), 481–490.

Pillonetto, G., Chen, T., Chiuso, A., De Nicolao, G., and Ljung, L. (2022). *Regularized System Identification*. Springer.

Pillonetto, G., Dinuzzo, F., Chen, T., De Nicolao, G., and Ljung, L. (2014). Kernel methods in system identification, machine learning and function estimation: A survey. *Automatica*, 50(3), 657–682.

Risuleo, R.S., Bottegal, G., and Hjalmarsson, H. (2019). Identification of linear models from quantized data: a midpoint-projection approach. *IEEE Transactions on Automatic Control*, 65(7), 2801–2813.

Scandella, M., Mazzoleni, M., Formentin, S., and Previdi, F. (2022). Kernel-based identification of asymptotically stable continuous-time linear dynamical systems. *International Journal of Control*, 95(6), 1668–1681.

Schölkopf, B., Herbrich, R., and Smola, A.J. (2001). A generalized representer theorem. In *International Conference on Computational Learning Theory*, 416–426.

Strijbosch, N. and Oomen, T. (2022). Iterative learning control for intermittently sampled data: Monotonic convergence, design, and applications. *Automatica*, 139, Article 110171.

Wu, C.F.J. (1983). On the convergence properties of the EM algorithm. *The Annals of Statistics*, 95–103.

#### Appendix A. PROOF OF LEMMA 4

**Proof.** The MAP estimator for  $\mathbf{c}$  is computed by

$$\begin{aligned} \hat{\mathbf{c}}_{\text{MAP}} &= \arg \max_{\mathbf{c} \in \mathbb{R}^N} (\log p(\mathbf{y}_{1:N}|\mathbf{c}) + \log p(\mathbf{c})) \\ &= \arg \max_{\mathbf{c} \in \mathbb{R}^N} \left( -\frac{N}{2} \log(2\pi\sigma^2) - \frac{\log \det(\pi \mathbf{K}^{-1}/\gamma)}{2} \right. \\ &\quad \left. + \sum_{i=1}^N \log \left[ \int_{\eta_i}^{\eta_i+h} e^{\frac{-1}{2\sigma^2} (z_i - \mathbf{K}_i^\top \mathbf{c})^2} dz_i \right] - \gamma \mathbf{c}^\top \mathbf{K} \mathbf{c} \right), \quad (\text{A.1}) \end{aligned}$$

where we have used the same derivation as for  $\ell(g)$  in Theorem 3 for computing the log-likelihood term. By comparing (A.1) to (7), we find that  $\hat{\mathbf{c}}$  in (7) is simply the MAP estimate of  $\mathbf{c}$  within the model in (11).  $\square$

#### Appendix B. PROOF OF THEOREM 6

**Proof.** Since the  $Q$  function provided by Lemma 5 is concave in  $\mathbf{c}$ , it is sufficient to obtain the point(s) which make the gradient of the objective function equal to zero. The gradient of  $Q(\mathbf{c}, \hat{\mathbf{c}}^{(j)}) - \gamma \mathbf{c}^\top \mathbf{K} \mathbf{c}$  is given by

$$\begin{aligned} &\frac{\partial}{\partial \mathbf{c}} (Q(\mathbf{c}, \hat{\mathbf{c}}^{(j)}) - \gamma \mathbf{c}^\top \mathbf{K} \mathbf{c}) \\ &= \frac{-1}{\sigma^2} \sum_{i=1}^N \int_{\eta_i}^{\eta_i+h} \mathbf{K}_i (\mathbf{K}_i^\top \mathbf{c} - z_i) p(z_i | y(i\Delta), \hat{\mathbf{c}}^{(j)}) dz_i - 2\gamma \mathbf{K} \mathbf{c}. \end{aligned}$$

Setting the gradient to zero yields the iterations

$$\hat{\mathbf{c}}^{(j+1)} = \left( \sum_{i=1}^N \mathbf{K}_i \mathbf{K}_i^\top + 2\gamma \sigma^2 \mathbf{K} \right)^{-1} \sum_{i=1}^N \mathbf{K}_i \tilde{z}_i^{(j)}, \quad (\text{B.1})$$

where we have defined the conditional mean  $\tilde{z}_i^{(j)}$  as

$$\tilde{z}_i^{(j)} = \int_{\eta_i}^{\eta_i+h} z_i p(z_i | y(i\Delta), \hat{\mathbf{c}}^{(j)}) dz_i, \quad (\text{B.2})$$

and where we have taken into account that

$$\int_{\eta_i}^{\eta_i+h} p(z_i | y(i\Delta), \hat{\mathbf{c}}^{(j)}) dz_i = 1, \quad i = 1, 2, \dots, N.$$

The iterations (14) are obtained from (B.1) by rewriting the sum related to  $\tilde{z}_i^{(j)}$  and using the fact that  $\sum_{i=1}^N \mathbf{K}_i \mathbf{K}_i^\top = \mathbf{K}^2$ , which holds since  $\mathbf{K}$  is symmetric. Finally, the explicit expression for  $\tilde{z}_i^{(j)}$  in (15) can be obtained directly from expanding the following alternative expression for (15) based on applying Bayes' theorem on the conditional expectation in (B.2):

$$\tilde{z}_i^{(j)} = \frac{\int_{\eta_i}^{\eta_i+h} z_i \exp \left( \frac{-1}{2\sigma^2} [z_i - \mathbf{K}_i^\top \hat{\mathbf{c}}^{(j)}]^2 \right) dz_i}{\int_{\eta_i}^{\eta_i+h} \exp \left( \frac{-1}{2\sigma^2} [z_i - \mathbf{K}_i^\top \hat{\mathbf{c}}^{(j)}]^2 \right) dz_i}. \quad \square$$

#### Appendix C. PROOF OF THEOREM 7

**Proof.** We seek to derive the MAP-EM iterations for computing the estimator in (16). By setting the latent variable as  $\mathbf{z}_{1:N}$ , we must compute the following  $Q$  function

$$Q(\boldsymbol{\rho}, \hat{\boldsymbol{\rho}}^{(j)}) = \mathbb{E} \left\{ \log p(\mathbf{z}_{1:N}, \mathbf{y}_{1:N} | \boldsymbol{\rho}) | \mathbf{y}_{1:N}, \hat{\boldsymbol{\rho}}^{(j)} \right\}.$$

It can be shown that (cf. Eq. (19) of Godoy et al. (2011))

$$p(\mathbf{z}_{1:N}, \mathbf{y}_{1:N} | \boldsymbol{\rho}) = \begin{cases} p(\mathbf{z}_{1:N} | \boldsymbol{\rho}) & \text{if } \mathbf{z}_{1:N} \in \mathbf{y}_{1:N}, \\ 0 & \text{otherwise,} \end{cases}$$

which, by exploiting (17), leads to

$$-2Q(\boldsymbol{\rho}, \hat{\boldsymbol{\rho}}^{(j)}) = \log \det(2\pi \mathbf{S}_\rho) + \mathbb{E} \{ \mathbf{z}_{1:N}^\top \mathbf{S}_\rho^{-1} \mathbf{z}_{1:N} | \mathbf{y}_{1:N}, \hat{\boldsymbol{\rho}}^{(j)} \}.$$

The iterations in (19) follow from applying the commutativity property of the trace to the expectation above.

The minimization of  $-2Q(\boldsymbol{\rho}, \hat{\boldsymbol{\rho}}^{(j)})$  with respect to  $\boldsymbol{\rho}$  provides the M-step of the MAP-EM iterations for computing a maximum of the likelihood of interest, which in turn is equivalent to solving the optimization problem in (18).  $\square$