

Obtaining Multivariable Continuous-Time Models From Sampled Data

Rodrigo A. Romano, Felipe Pait and P. Lopes dos Santos

Abstract—While most physical systems or phenomena occur in continuous-time, identification methods based on discrete-time models are more widespread among practitioners and academic community, possibly due to the discrete-time nature of the data records. There has been a growing interest in estimating continuous-time (CT) models in the last decade. This work develops algorithms to estimate the parameters of multivariable state-space CT models from input-output samples using a method based on the recently developed MOLI-ZOFT approach. The performance of the algorithm is evaluated using real data from an industrial winding process.

I. INTRODUCTION

In a recent contribution [12], identification of linear time-invariant multivariable input-output systems is approached by solving three separate problems. First is selection of a structure containing system models capable of approximating the input-output behavior of the process to be identified. Second is the design of filters that separate process signals from disturbances and measurement noise, in a manner conducive to parameter estimation. Third is parameter estimation itself. The method was dubbed MOLI-ZOFT, for Matchable-Observable Linear Identification with Zero-order Oracle Filter Tuning.

The approach makes use of MIMO quasi-canonical forms, which lead to regressor-form observers for the process models presented in Section II. Originally described in continuous-time in the context of adaptive control [9], they are applied to discrete-time system identification in [12]. Their salient property of being based on system representations with fixed dynamics, where the unknown parameters appear in the readout matrix only, makes them eminently suitable for identification of continuous-time systems using sampled data only.

Reasons to identify continuous-time models from sampled data include: continuous-time models are more familiar and convenient for feedback control design; achievable and desirable sampling rates for data collection and for feedback control may differ; and unless sampling is periodic, a time-invariant system appears time-varying in discrete time. These issues are elaborated in detail in a recent survey [4] and its references. To the best of our knowledge, identification of multivariable continuous-time system from sampled data lags behind both its discrete-time and SISO counterparts. The

methods presented here are applicable to MIMO and to SISO systems, and offer advantages in both cases.

The parametrized models used are described in Section II. The estimation algorithms, which furnish a continuous-time model directly from the sampled data, rather than indirectly by first estimating discrete-time dynamics parameters, are in Section III. Experimental results are discussed in Section IV and a conclusion in Section V.

II. MODEL PARAMETERIZATION

We tackle the multivariable linear system identification problem using the following continuous-time model structure

$$\dot{x}(t) = A_m x(t) + B_m u(t) \quad (1)$$

$$y(t) = C_m x(t) + D_m u(t), \quad (2)$$

with $x \in \mathbb{R}^{n_x}$, $y \in \mathbb{R}^{n_y}$, $u \in \mathbb{R}^{n_u}$ and

$$A_m = A + L(\theta) (I_{n_y} - H(\theta))^{-1} C$$

$$B_m = B(\theta) + L(\theta) (I_{n_y} - H(\theta))^{-1} D(\theta)$$

$$C_m = (I_{n_y} - H(\theta))^{-1} C$$

$$D_m = (I_{n_y} - H(\theta))^{-1} D(\theta),$$

where θ denotes the model parameters, $I_{n_y} \in \mathbb{R}^{n_y \times n_y}$ is an identity matrix, and (C, A) is a user-defined observable pair such that A is stable (that is, all its eigenvalues have strictly negative real parts). The parameter matrix $H(\theta) \in \mathbb{R}^{n_y \times n_y}$ is strictly lower triangular. The other parameter matrices $L(\theta)$, $B(\theta)$ and $D(\theta)$ take values in $\mathbb{R}^{n_x \times n_y}$, $\mathbb{R}^{n_x \times n_u}$ and $\mathbb{R}^{n_y \times n_u}$, respectively. Therefore, the dimension of the parameter vector is

$$\dim \theta = \frac{n_y}{2} (n_y - 1) + n_x (n_y + n_u) + n_y n_u.$$

which is not minimal, but considerably more parsimonious than fully-parameterized structures, commonly used in subspace identification algorithms.

The pair (C, A) can be constructed as follows. Let $l_o = \{n_1, \dots, n_{n_y}\}$ be a list of n_y positive integers satisfying $n_1 + \dots + n_{n_y} = n_x$ (the elements of the list l_o are known as the observability indices of the pair (C, A)). First, adopt a stable monic polynomial $\alpha(s)$ of degree $\underline{n} = \max(l_o)$, namely

$$\alpha(s) = s^{\underline{n}} + \alpha_1 s^{\underline{n}-1} + \dots + \alpha_{\underline{n}-1} s + \alpha_{\underline{n}},$$

such that, $\alpha(s)$ has a real monic factor $\gamma_i(s)$ of degree n_i , for each $i = \{1, \dots, n_y\}$. Then, matrices C and A are given by

$$C = \text{block diagonal} \{c_1, \dots, c_{n_y}\} \quad (3)$$

$$A = \text{block diagonal} \{A_1, \dots, A_{n_y}\}, \quad (4)$$

R. A. Romano is with Escola de Engenharia Mauá, Instituto Mauá de Tecnologia, São Caetano do Sul, SP, Brazil; e-mail: rromano@maua.br.

F. Pait is with Escola Politécnica da Universidade de São Paulo, SP, Brazil; e-mail: pait@lac.usp.br.

P. Lopes dos Santos is with Faculdade de Engenharia da Universidade do Porto, Portugal; email: pjsantos@fe.up.pt.

where (c_i, A_i) are n_i -dimensional observable pairs, for which $\gamma_i(s)$ is the characteristic polynomial of each A_i .

The parameterization (1)–(2) is capable of matching any $n_y \times n_u$ transfer matrices with a list of observability indices l_o , and it is observable for all θ [9]. As Section II-A will show, this model structure is particularly suitable for parameter estimation using linear least-squares.

A. Regression form

Consider the \underline{n} -dimensional observable pair $(\underline{c}, \underline{A})$ such that $\alpha(s)$ is the characteristic polynomial of \underline{A} . As $\gamma_i(s)$ is a factor of $\alpha(s)$, there are left-invertible matrices $M_i \in \mathbb{R}^{\underline{n} \times n_i}$ that extract an observable n_i -dimensional subspace from the observability matrix constructed from the pair $(\underline{c}, \underline{A})$. This leads to

$$c_i (sI_{n_i} - A_i)^{-1} = \underline{c} (sI_{\underline{n}} - \underline{A})^{-1} M_i. \quad (5)$$

As shown in [12, Lemma 1] for $(c_i, A_i) = (c_i^*, A_i^*)$, where $c_i^* = [0 \ \dots \ 0 \ 1]$ and A_i^* is in right companion form, matrices M_i are equal to

$$M_i^* = \begin{bmatrix} \kappa_{i,1} & 0 & \dots & 0 \\ \vdots & \kappa_{i,1} & & \vdots \\ \kappa_{i,\underline{n}-n_i} & \vdots & \ddots & 0 \\ 1 & \kappa_{i,\underline{n}-n_i} & & \kappa_{i,1} \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \kappa_{i,\underline{n}-n_i} \\ 0 & \dots & 0 & 1 \end{bmatrix},$$

whose entries are given by the coefficients of the polynomial

$$\begin{aligned} \kappa_i(s) &= \frac{\alpha(s)}{\gamma_i(s)} \\ &= s^{\underline{n}-n_i} + \kappa_{i,\underline{n}-n_i} s^{\underline{n}-n_i-1} + \dots + \kappa_{i,2} s + \kappa_{i,1}. \end{aligned}$$

There is no loss of generality in making $c_i = c_i^* T_i$ and $A_i = T_i^{-1} A_i^* T_i$, where T_i are suitable transformation matrices. Then, using (5) one achieves

$$M_i = \underline{T}^{-1} M_i^* T_i,$$

where \underline{T} is such that $\underline{A} = \underline{T}^{-1} \underline{A}^* \underline{T}$, for a right companion matrix \underline{A}^* .

Next, for each $i \in \{1, \dots, n_y\}$, define the parameter vector

$$\theta_i = \left[\text{vec}([L_i(\theta) \ B_i(\theta)])^\top \ h_{i1} \ \dots \ h_{i(i-1)} \ \varrho_i D(\theta) \right]^\top,$$

where the operator $\text{vec}(\cdot)$ stacks the columns of the argument on top of each other. Letting $\sigma_0 = 0$, $\sigma_i = n_i + \sigma_{i-1}$, the matrices L_i and B_i are the respective partitions of L and B formed of the block from row $\sigma_{i-1} + 1$ to σ_i . The symbols h_{ij} denote the nonzero elements of the matrix $H(\theta)$ and the vector $\varrho_i \in \mathbb{R}^{n_y}$ is filled with zeros, except for the i th entry, which is set to 1.

Proposition 1: Let

$$\begin{aligned} \mathcal{M}_i &= I_{n_u+n_y} \otimes M_i \\ \mathcal{A} &= I_{n_u+n_y} \otimes \underline{A}^\top \\ \mathcal{C} &= I_{n_u+n_y} \otimes \underline{c}^\top, \end{aligned} \quad (6)$$

where the symbol \otimes denotes the Kronecker product. The i th output prediction based on model (1)–(2) can be expressed in the regression form

$$\hat{y}_i(t) = \varphi_i(t)^\top \theta_i, \quad (7)$$

where the corresponding regression¹ vector is

$$\varphi_i(t) = [\xi(t_k)^\top \mathcal{M}_i \ y_1(t) \ \dots \ y_{i-1}(t) \ u(t)^\top]^\top \quad (8)$$

and $\xi(t)$ is the response of the $\underline{n}(n_u + n_y) \times (n_u + n_y)$ -dimensional transfer function matrix

$$\Sigma(s) = (sI_{\underline{n}(n_u+n_y)} - \mathcal{A})^{-1} \mathcal{C} \quad (9)$$

to the input $z(t) = [y(t)^\top \ u(t)^\top]^\top$.

Proof: The block diagonal form of C and A enables (1) to be decomposed into n_y subsystems whose state equations read

$$\dot{x}_i(t) = A_i x_i(t) + L_i(\theta) y(t) + B_i(\theta) u(t),$$

where $x_i \in \mathbb{R}^{n_i}$ is the i th partition of partition of

$$x = [x_1^\top \ \dots \ x_{n_y}^\top]^\top.$$

From (2) and (5), the i th model output prediction is given by

$$\begin{aligned} \hat{y}_i(t) &= \underline{c} (sI_{\underline{n}} - \underline{A})^{-1} M_i (L_i(\theta) y(t) + B_i(\theta) u(t)) \\ &\quad + \varrho_i (H(\theta) y(t) + D(\theta) u(t)). \end{aligned} \quad (10)$$

Here s represents the derivative operator d/dt , borrowing the same letter used to denote the Laplace transform variable. Applying the $\text{vec}(\cdot)$ operator to the first term in the right hand side of the previous equation yields

$$\begin{aligned} &\text{vec} \left(\underline{c} (sI_{\underline{n}} - \underline{A})^{-1} M_i [L_i(\theta) \ B_i(\theta)] z(t) \right) \\ &= \text{vec} \left(\sum_{l=0}^{\infty} s^{-(1+l)} \underline{c} \underline{A}^l M_i [L_i(\theta) \ B_i(\theta)] z(t) \right) \\ &= \sum_{l=0}^{\infty} s^{-(1+l)} \left(z(t)^\top \otimes \underline{c} \underline{A}^l M_i \right) \text{vec}([L_i(\theta) \ B_i(\theta)]). \end{aligned}$$

The summation in the previous equation can be rewritten as

$$\begin{aligned} &\sum_{l=0}^{\infty} s^{-(1+l)} (z(t)^\top \otimes \underline{c}) (I_{n_u+n_y} \otimes \underline{A}^l M_i) \\ &= \mathcal{M}_i^\top (sI_{n_u+n_y} - \mathcal{A})^{-1} (z(t) \otimes \underline{c}^\top) \\ &= \mathcal{M}_i^\top (sI_{n_u+n_y} - \mathcal{A})^{-1} \mathcal{C} z(t) \\ &= \mathcal{M}_i^\top \xi(t). \end{aligned}$$

Thus,

$$\begin{aligned} &\text{vec} \left(\underline{c} (sI_{\underline{n}} - \underline{A})^{-1} M_i [L_i(\theta) \ B_i(\theta)] z(t) \right) \\ &= \text{vec}([L_i(\theta) \ B_i(\theta)])^\top \mathcal{M}_i^\top \xi(t). \end{aligned} \quad (11)$$

¹For $i = 1$, the parameter and the regressor vector is

$$\begin{aligned} \theta_1 &= [\text{vec}([L_1(\theta) \ B_1(\theta)])^\top \ \varrho_1 D(\theta)]^\top \\ \varphi_1(t) &= [\xi(t_k)^\top \mathcal{M}_1 \ u(t)^\top]^\top. \end{aligned}$$

and substituting (11) into (10) results in

$$\hat{y}_i(t) = \xi(t)^\top \mathcal{M}_i \text{vec}([L_i(\theta) \ B_i(\theta)]) + \varrho_i(H(\theta)y(t) + D(\theta)u(t))$$

which is equivalent to (7). ■

Remark: The regressors in (7) incorporate input–output data, and the $\underline{n}(n_u + n_y)$ –dimensional signal vector $\xi(t)$, computed using the transfer function matrix $\Sigma(s)$, defined in (9), whose poles are the eigenvalues of \underline{A} . Derivatives of the input and output signals are not computed explicitly in our formulation, but there is a connection between the role of the polynomial $\alpha(s)$ and the design parameter used to adjust the bandwidth of the traditional state variable filters (SVF) [6]. Hence, as in the SVF approach, the poles of $\Sigma(s)$ can be designed to remove undesirable high frequency content (disturbances) from the estimation data. Naturally, “optimal” filter tuning require the knowledge of the true plant dynamics and of the noise/disturbance spectrum, which is unavailable in practice. A data–driven, derivative–free optimization procedure to tune the polynomial $\alpha(s)$ is presented in Section III-B.

III. ESTIMATION ALGORITHMS

A. Parameter estimation

Firstly, suppose that the polynomial $\alpha(s)$ is chosen beforehand. Given the list l_o , it is straightforward to construct φ_i as defined in (8) from $y(t)$, $u(t)$ and $\xi(t)$ subject to

$$\dot{\xi}(t) = \mathcal{A}\xi(t) + \mathcal{C}z(t). \quad (12)$$

Intersample behavior of the observed data will be assumed or extrapolated, unless additional information is available. Then, using the regression model (7) at time instants $t = t_k$ (for $k = 1, \dots, N$) the parameter estimation problem is formulated as

$$\hat{\theta}_i = \arg \min_{\theta_i} \sum_{k=1}^N (y(t_k) - \varphi_i(t_k)^\top \theta_i)^2, \quad (13)$$

which can be efficiently solved for each $i \in \{1, \dots, n_y\}$ by means of linear least–squares methods.

The step–by–step procedure to estimate the parameters of the continuous–time model (1)–(2) is summarized in Algorithm 1.

Remark: The procedure described in Algorithm 1 assumes that the list of observability indices l_o is known *a priori*. In a “gray–box” setting (see e.g., [8]), this structural information may be retrieved from first–principle models. In the case of a “black–box” approach, it is still possible to pick l_o using some structure selection method, e.g. [7], [13].

Algorithm 1 (CT–MOLI) Parameter estimation of continuous–time models from sampled data

input: A dataset $\mathcal{Z} = \{y(t_1), u(t_1), \dots, y(t_N), u(t_N)\}$, l_o , $\alpha(q)$ and its factors, namely γ_i .
 Evaluate $\xi(t)$ subject to (12) at $t = t_k$, $k = 1, \dots, N$.
for $i = 1$ **to** n_y **do**
 Construct \mathcal{M}_i according to (6).
 Build $\varphi_i(t_k)$ as in (8).
 Solve θ_i in (13).
end for
 Compute A_m , B_m , C_m and D_m of model (1)–(2).

B. Filter tuning

Previously it was pointed out that the eigenvalues of the user defined matrix \underline{A} determine the poles of $\Sigma(s)$, which generate $\xi(t)$ from the input–output data. Therefore, the choice of the eigenvalues of \underline{A} can be interpreted as a filter design. Rather than treating all the coefficients of the characteristic polynomial as free design parameters, we parametrize $\alpha(s)$ using Butterworth polynomials:

$$\alpha(s) = \prod_{m=1}^{\underline{n}} (s - s_m), \quad (14)$$

where

$$s_m = \omega_c \exp\left(j \frac{(2m + \underline{n} - 1)}{2\underline{n}}\right).$$

The coefficients of $\alpha(s)$ are expressed as a function of the cutoff frequency ω_c , which has a clear physical interpretation. Filter tuning becomes a less complex search in a subspace of dimension 1, at the price of less flexibility in the frequency response function shape of the filter.

According to the MOLI–ZOFT approach [12], in this work ω_c is tuned using a derivative–free optimization method. With this aim, we consider a set $\Omega = \{\omega_1, \dots, \omega_{n_\omega}\}$ of candidate values of ω_c called “curiosity points”, and let $J(\omega_c, \mathcal{Z})$ be a functional which quantifies the performance of the values $\omega_c \in \Omega$ given a data set

$$\mathcal{Z} = \{y(t_1), u(t_1), \dots, y(t_N), u(t_N)\}.$$

Then, the filter cutoff frequency is calculated through

$$\omega_c^* = \frac{\sum_{v=1}^{n_\omega} \omega_v e^{-\mu J(\omega_v, \mathcal{Z})}}{\sum_{v=1}^{n_\omega} e^{-\mu J(\omega_v, \mathcal{Z})}}. \quad (15)$$

Therefore, ω_c^* is the barycenter of the curiosity points ω_v weighted by the term $e^{-\mu J(\omega_v, \mathcal{Z})}$. The positive constant μ is used to adjust weighting terms — the higher μ , the more ω_c^* tends to the element in Ω that provides the lowest J (best performance). The rationale behind (15) is that curiosity points which achieve better performance are given more weight than those that lead to worse results.

The barycenter can be seen as a direct optimization method because only the numerical values of $J(\omega_v, \mathcal{Z})$ have to be computed for each $v \in \{1, \dots, n_\omega\}$, and its derivatives are not required. Considerable freedom is retained in the choice of the functional J . This flexibility can be used to reflect the

model application; in Section IV we used one based on the normalized root mean square error (NRMSE). Alternatives to the barycenter method, which was selected for its simplicity, effectiveness and robustness [11], [10], are described in the derivative-free optimization literature [3].

The combined algorithm is summarized below.

Algorithm 2 (CT-MOLI-ZOFT) Parameter estimation coupled with data-driven filter tuning

input: A dataset \mathcal{Z} , l_o , Ω and γ_i .

for $v = 1$ **to** n_ω **do**

Build the polynomial $\alpha(s)$ using (14) with $\omega_c = \omega_v$.

Estimate the parameters of model (1)–(2) using \mathcal{Z} , l_o , $\alpha(s)$ and its factors γ_i as inputs of Algorithm 1.

Evaluate the functional $J(\omega_v, \mathcal{Z})$.

end for

Compute the barycenter (15).

Build the polynomial $\alpha(s)$ using (14) with $\omega_c = \omega_v$.

Estimate the parameters of model (1)–(2) using \mathcal{Z} , l_o , $\alpha(s)$ and its factors γ_i as inputs of Algorithm 1.

Remark: If there is prior knowledge about the frequency range of the plant dynamics, such information can be easily incorporated through the determination of the elements that form Ω . On the other hand, in case of the unavailability of any information, it is possible to adopt a lower and an upper bound for the bandwidth, and then test different cutoff frequencies spread in this range. This strategy is used in the next section.

IV. EXPERIMENTAL RESULTS

The proposed estimation method is assessed using real data from an industrial winding process. This multivariable plant is composed of a plastic web that is unwound from a first reel, goes over the traction reel and is finally rewinded on the the last reel. Reels 1 and 3 are driven by DC motors, which are controlled with set-point currents I_1^* and I_3^* , respectively. Another control loop regulates the angular speed of the reel 2 (S_2), whose set-point is denoted by S_2^* (see Figure 1, which appears in [2]). Essentially, the role of a winding process is to control the web linear velocity and the web tensions around a given operating point.

The experimental dataset is available from the CONTSID (CONtinuous-Time System IDentification) toolbox² [5], in which the winding process identification is presented as a MIMO case-study in example file `idcdemo7.m`. The manipulated input vector is composed of

$$u(t) = [I_1^*(t) \quad S_2^*(t) \quad I_3^*(t)]^\top$$

and the process output signal vector is

$$y(t) = [T_1(t) \quad T_3(t) \quad S_2(t)]^\top,$$

²The CONTSID toolbox can be freely downloaded from: www.contsid.cran.univ-lorraine.fr

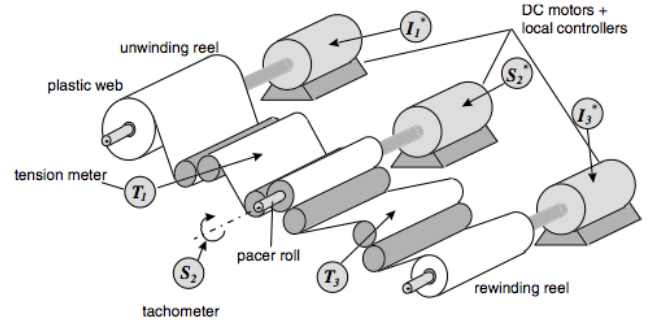


Fig. 1. Winding process schematic [2].

where T_1 and T_3 are the tensions between reels 1–2 and between reels 2–3, respectively.

The plant was excited in open-loop using binary signals during 60s and the data was recorded using a sampling period of $T = 0.01$ s. The experimental data was split in two halves. The first is used for parameter estimation and the second one is employed in the model validation.

In order to focus on the assessment of parameter estimation and filter tuning algorithms, the model order is assumed to be $n_x = 3$, the same as the one encountered in a previous study about winding process identification using continuous-time models [1]. For the sake of comparison, we also considered the results provided by the PMF-based subspace algorithm (4SID-PMF) described in the same work, and implemented in the CONTSID toolbox. The acronym PMF stands for Poisson Moment Function, which is the method used to reconstruct the time derivatives of the input–output data.

The *best fit rate* (BFR_i) was used to evaluate the models accuracy. This index is defined as

$$BFR_i(\%) = 100 \cdot \max \left(1 - \frac{\|y_i - y_i^s\|_2}{\|y_i - \bar{y}\|_2}, 0 \right),$$

where $\|\cdot\|_2$ is the ℓ_2 norm of the argument, y_i^s is the i th output simulated by the estimated model and \bar{y} is the mean of the i th observed output sequence y_i .

We have adopted $l_o = \{1, 1, 1\}$ in Algorithms 1 and 2 — the only possible values of the observability indices for a third order model with $n_y = 3$ without any output with index 0. In the first, the cutoff frequency is set to $\omega_c = 4.25$ rad/s, which corresponds to the bandwidth that provided the best performance for 4SID-PMF in [1]. For Algorithm 2 the set of curiosities comprised seven values logarithmically spaced between 0.2 and 6rad/s. The barycenter (15) is calculated using the index of merit

$$J(\omega_v, \mathcal{Z}) = \frac{1}{n_y} \sum_{i=1}^{n_y} \left(\frac{\|y_i - y_i^s\|_2}{\|y_i - \bar{y}\|_2} \right).$$

The barycenter ω_c^* as well as the index of merit achieved by each candidate value in Ω are shown in Figure 2. The behavior of the functional $J(\omega_v, \mathcal{Z})$ reveals that our estimation method performs better for cutoff frequencies around 1.6rad/s; the barycenter is in 1.572rad/s. Although the parameter used

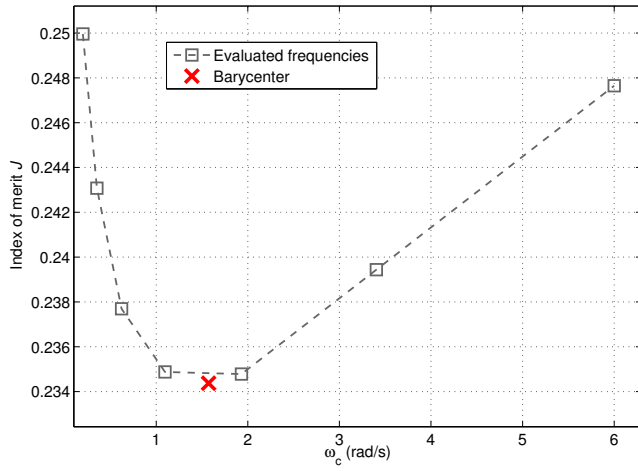


Fig. 2. Data-driven filter tuning using direct optimization.

to adjust the filter bandwidth in the PMF approach plays a similar role to the cutoff frequency used to parameterize $\alpha(s)$, our formulation has significant structural differences³. Hence, these values are by no means the same, as evidenced by the previous result.

The simulated outputs reproduced by the estimated models and the actual process outputs are depicted in Figure 3, which shows a good match between the estimates and the validation data. The BFR computed for each output using the validation dataset is presented in Table I. Notice that even if ω_c is set to 4.25rad/s, Algorithm 1 (CT-MOLI) fits better than 4SID-PMF in all outputs. Moreover, if the filter is tuned using the barycenter, as in Algorithm 2 (CT-MOLI-ZOFT) the estimated model achieves an improved fit in every output.

The computational load of each estimation method was also measured. The last column in Table I reports that CT-MOLI is quite fast, while CT-MOLI-ZOFT obtains more precise results in approximately the same time as a single run of 4SID-PMF algorithm. The computations were done using an iMac with 2.7GHz Intel Core i5 processor and 8GB memory.

V. CONCLUSION

We have established that the techniques recently developed for discrete-time system identification by the authors can be usefully applied to the estimation of continuous-time multivariable models, with good results from both the point of view of precision and computational complexity. The method is particularly well-suited for grey-box system identification; work is underway and will be reported shortly. The methods described here will also be used for the identification of switched systems.

³For this case-study the choice $l_o = \{1, 1, 1\}$ leads to a set of six first-order filters to generate $\xi(t)$, while optimal results for the 4SID-PMF are attained when each time derivative of the input-output data are computed using the filters

$$f_{\iota,j} = \frac{\beta^{j+1} s^\iota}{(s + \lambda_{\text{PMF}})^{j+1}},$$

with $\iota = \{0, 1, 2\}$, $j = 2$ and $\lambda_{\text{PMF}} = 4.25\text{rad/s}$ [1].

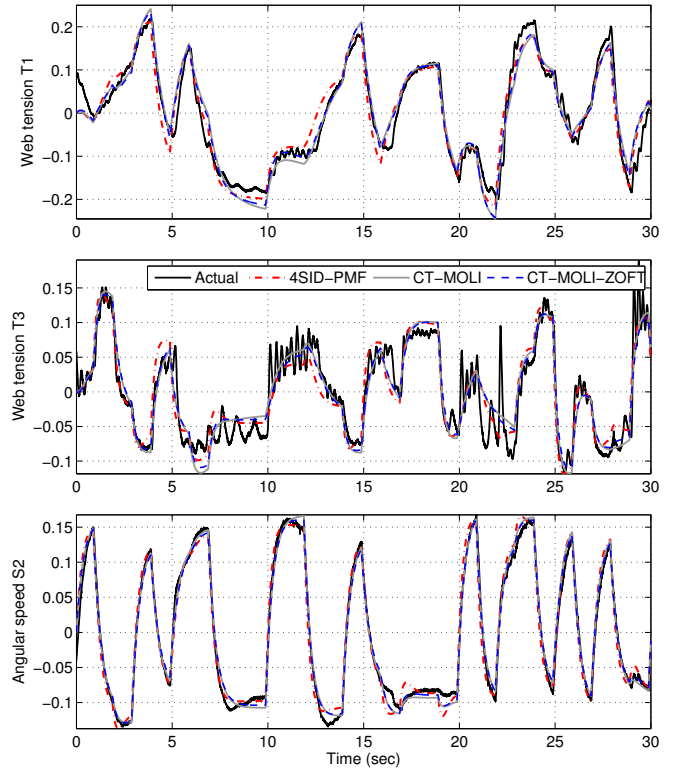


Fig. 3. Cross validation.

TABLE I
BFR COMPUTED FROM THE VALIDATION DATA AND COMPUTATION TIME
OF THE ESTIMATION METHODS.

	BFR			runtime
	y_1	y_2	y_3	
4SID-PMF	72.2%	59.3%	84.3%	167ms
CT-MOLI	73.8%	63.1%	88.8%	27.2ms
CT-MOLI-ZOFT	75.7%	63.5%	89.3%	170ms

ACKNOWLEDGMENT

The authors gratefully acknowledge the support of this research by Instituto Mauá de Tecnologia (IMT). We thanks Hugues Garnier for kindly making available the CONTSID toolbox.

REFERENCES

- [1] T. Bastogne, H. Garnier, and P. Sibille, "A PMF-based subspace method for continuous-time model identification. application to a multivariable winding process," *International Journal of Control*, vol. 74, no. 2, pp. 118–132, 2001.
- [2] T. Bastogne, H. Noura, P. Sibille, and A. Richard, "Multivariable identification of a winding process by subspace methods for tension control," *Control Engineering Practice*, vol. 6, pp. 1077–1088, 1998.
- [3] A. R. Conn, K. Scheinberg, and L. N. Vicente, *Introduction to Derivative-Free Optimization*, ser. MPS-SIAM series on optimization. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2009.
- [4] H. Garnier, "Direct continuous-time approaches to system identification. Overview and benefits for practical applications," *European Journal of Control*, vol. 24, pp. 50–62, 2015.

- [5] H. Garnier, M. Gilson, T. Bastogne, and M. Mensler, *Identification of Continuous-time Models from Sampled Data*. London: Springer-Verlag, 2008, ch. The CONTSID Toolbox: A Software Support for Data-based Continuous-time Modelling, pp. 249–290.
- [6] H. Garnier, M. Mensler, and A. Richard, “Continuous-time model identification from sampled data: implementation issues and performance evaluation,” *International Journal of Control*, vol. 76, no. 13, pp. 1337–1357, 2003.
- [7] R. Guidorzi, “Canonical structures in the identification of multivariable systems,” *Automatica*, vol. 11, pp. 361–374, 1975.
- [8] G. Mercère, O. Prot, and J. A. Ramos, “Identification of parameterized gray-box state-space systems: From a black-box linear time-invariant representation to a structured one,” *IEEE Transactions on Automatic Control*, vol. 59, no. 11, pp. 2873–2885, Nov 2014.
- [9] A. S. Morse and F. M. Pait, “MIMO design models and internal regulators for cyclicly switched parameter-adaptive control systems,” *IEEE Transactions on Automatic Control*, vol. 39, no. 9, pp. 1809–1818, 1994.
- [10] F. Pait, “Reading Wiener in Rio,” in *IEEE Conference on Norbert Wiener in the 21st Century*. IEEE, 2014, pp. 1–4.
- [11] F. Pait and D. Colón, “A barycenter method for direct optimization,” in *SIAM Conference on Optimization*, San Diego, USA, May 2014.
- [12] R. A. Romano and F. Pait, “Matchable-observable linear models and direct filter tuning: An approach to multivariable identification,” *IEEE Transactions on Automatic Control*, 2017, DOI: 10.1109/TAC.2016.2602891.
- [13] R. A. Romano, F. Pait, and R. C. Ferrão, “Matchable-observable linear models for multivariable identification: Structure selection and experimental results,” in *Proceedings of the 54th IEEE Conference on Decision and Control*, Osaka, 2015, pp. 3391–3396.