On Data Segmentation for Switched Linear Models

Felipe Pait, *Senior Member, IEEE*, and Rodrigo A. Romano, *Member, IEEE*

Abstract— **Estimation of parameters and switching times for linear systems is an important problem that can pose significant computational challenges. Using matchable–observable linear models which have recently been employed in discrete–time system identification, we approach the problem by splitting the overall task into simpler, well–defined problems. Treating the estimation and segmentation tasks separately allows one to define cost functions which depend on the switching times, so that segmentation can be formulated as a nonlinear optimization problem. The goal functions may not be convex or they may have other properties that can cause some standard nonlinear programming algorithms to fail and others to perform poorly. Nevertheless, approximate considerations of the properties of the said functions shows that the optimization problems can be approached by derivative–free methods, including simple bisection. Qualitative analysis indicates that, with suitable model structures, the approach leads to efficient algorithms. The reason is that the separate tasks can each be approached using effective optimization techniques, while the combined problem of parameter estimation and segmentation would require challenging nonconvex optimizations. Once segmentation is accomplished, estimation of the dynamical system parameters themselves become a matter of employing standard identification techniques. Numerical simulations confirm the potential of the method proposed.**

I. A MULTIVARIABLE HYBRID LINEAR MODEL

We wish to estimate the parameters of a multivariable hybrid linear model

$$\dot{x}(t) = A_\sigma^M x(t) + B_\sigma^M u(t) \tag{1}$$

$$y(t) = C_\sigma^M x(t) + D_\sigma^M u(t), \tag{2}$$

using sampled measurements $\{u(t_k), y(t_k)\}$, $k \in \{0, 1, \ldots, N\}$ of the input and output signals. The switching signal $\sigma(t) \in \{1, 2, \ldots, S\}$ is piecewise constant between the model switching times $\Upsilon = \{\tau_1, \ldots, \tau_S\}$, which themselves are unknown a priori and need to be determined, and $x \in \mathbb{R}^{n_x}$, $y \in \mathbb{R}^{n_y}$, $u \in \mathbb{R}^{n_u}$.

Segmentation of data from N sequential samples into S linear models is useful and important in diverse applications, but can pose a significant challenge because of the computational complexity of the combined problem of classifying models and estimating parameters and switching times [9]. We approach the problem by splitting the overall task into a number of simpler, well–defined problems.

Using MOLI models which have been employed in discrete–time system identification in [11], MIMO estimation can be treated with little additional complexity with respect to the SISO case. Likewise estimation of continuous–time models from sampled data poses no great difficulty.

Reasons for describing the models in continuous time, while employing sampled measurements, are discussed in [12] and elaborated in detail in a recent survey [3] and its references. We would add, as a matter of personal taste, that in the opinion of the authors the conceptual transparency offered by the separation between the dynamics and the switched behaviors in the continuous–time framework makes the developments presented here significantly more intuitive. For the problem at hand, perhaps the strongest argument for the development in continuous time is that it equips us for the approximate, qualitative analysis in Section III, which in turn suggests which optimization algorithms will be most effective for joint segmentation and estimation.

The model parametrization is detailed in Section II. The results in Section III form a conceptual basis for the algorithmic developments that follows, and serve as an illustration of the advantages offered by continuous–time thinking in the present context. In Section IV we describe the algorithms used for the simulation tests in Section V.

II. MODEL PARAMETERIZATION

In this work, we consider linear state–space models in which the parameter matrices in (1)–(2) assume the following structure

$$\begin{aligned}
A_\sigma^M(t) &= A + L(\theta_\sigma)\left(I_{n_y} - G(\theta_\sigma)\right)^{-1} C \\
B_\sigma^M(t) &= B(\theta_\sigma) + L(\theta_\sigma)\left(I_{n_y} - G(\theta_\sigma)\right)^{-1} D(\theta_\sigma) \\
C_\sigma^M(t) &= \left(I_{n_y} - G(\theta_\sigma)\right)^{-1} C \\
D_\sigma^M(t) &= \left(I_{n_y} - G(\theta_\sigma)\right)^{-1} D(\theta_\sigma),
\end{aligned}$$

where I_{n_y} is an identity matrix of dimension n_y, (C, A) is an user–defined observable pair, such that A is stable. The model parameters θ_σ[1] are the entries of the strictly lower triangular matrix $G(\theta_\sigma) \in \mathbb{R}^{n_y \times n_y}$ and of the matrices $L(\theta_\sigma)$, $B(\theta_\sigma)$ and $D(\theta_\sigma)$ that 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.

The rationale behind the use of this model structure, which was fully spelled out in [6] (with basis on previous work on classical realization theory [5], [2]), and put to use in system identification in [11], lies in that the models are capable of matching all processes with their linear system structure. In addition, this state–space parameterization admits an output

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

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

[1] The time–dependence of the parameter vector is dropped to simplify notation.

Preprint submitted to 57th IEEE Conference on Decision and Control.
Received March 20, 2018.

predictor of the form

$$\dot{x}_{\mathcal{O}}(t) = A_{\mathcal{O}} x_{\mathcal{O}}(t) + D_{\mathcal{O}} y(t) + B_{\mathcal{O}} u(t) \tag{3}$$

$$\hat{y}(t) = C_{\mathcal{O}}(\theta_\sigma) x_{\mathcal{O}}(t) + G(\theta_\sigma) y(t) + D(\theta_\sigma) u(t). \tag{4}$$

The construction of (3)–(4) begins with the choice of a single–output, observable pair $(\underline{c}, \underline{A})$ of dimension $\underline{n}$ equal to the largest element of the list of observability indices of the process model (1)–(2) ($\underline{n}$ is sometimes called the observability index of the system). The matrices $D_{\mathcal{O}}$ and $B_{\mathcal{O}}$ are composed of 0s and 1s, and

$$A_{\mathcal{O}} = I_{(n_y+n_u)} \otimes \underline{A}^T,$$

where $\otimes$ denotes the Kronecker product. The detailed construction of the predictor is omitted from this paper for reasons of space. For a more detailed explanation, including the mapping of the elements in matrices $L(\theta_\sigma)$ and $B(\theta_\sigma)$ of (1)–(2) into $C_{\mathcal{O}}(\theta_\sigma)$, we refer the reader to [6], [11].

We shall call realization (3)–(4) a Morse observer. It possesses two important features. First, the dynamics of its state vector $x_{\mathcal{O}}$ is set by the eigenvalues of $\underline{A}$, which are design variables that can be tuned to filter undesirable frequency content in the input and output signals. Second, define the information vector

$$\phi(t) = \begin{bmatrix} x_{\mathcal{O}}^T(t) & y^T(t) & u^T(t) \end{bmatrix}^T \in \mathbb{R}^{n_\phi},$$

where $n_\phi = \underline{n}(n_y + n_u) + n_y + n_u$. The model parameters

$$\theta_\sigma = \begin{bmatrix} C_{\mathcal{O}} & G & D \end{bmatrix} \in \mathbb{R}^{n_\phi \times n_y}$$

appear linearly with respect to ϕ, only in the output prediction equation. This enables us to write an output prediction based on model (1)–(2) in regression form

$$\hat{y}(t) = \theta_\sigma^T \phi(t). \tag{5}$$

III. The Smooth Shape of Things to Come

Mindful though we are of the need to obtain estimates of parameters and switching times on the basis of sampled data, in this section we consider integral functionals and their extrema. During this worthwhile detour, continuity permits the use of classical optimality conditions, which inform the subsequent choice of algorithms used to determine the parameters.

Three important points need to be emphasized here. First, the approximations developed in this section make sense for stable processes whose time constants in each operating mode is significantly shorter than the duration of the dwell time in each hybrid mode. We are not trying to obtain experimentally models for a system that switches as fast as the continuous dynamics.

Second, the calculations presented in this section are only approximate, and the values and functions obtained will *not* be used in determining the parameters of the switching models. What they are useful for is to exhibit the shape of the functions we propose to optimize as part of our estimation procedure, as detailed in Section IV. Knowledge about the shape of cost functionals, as illustrated in Figures 1 and 2, tells us that the overall problem of jointly estimating parameters and switching times is an exceedingly challenging nonconvex optimization problem. On the other hand, the parameter estimation task itself can be approached via least–squares methods, and the segmentation problem alone, although nonconvex, is amenable to simple treatment via, for example, naive bisection methods, or perhaps the recently developed barycenter method for derivative–free optimization reported in [10]. The combination of methods is what allows us to obtain numerical results that are orders of magnitude faster then other techniques previously studied in the literature, as will be seen in Section V.

Third, the development here contemplates the noise–free case, but as mentioned previously the regressor ϕ is the output of an observer which by construction acts as a filter on the noise. As the calculations are only approximate and used exclusively as a guide to pick optimization algorithms, imprecision in this section doesn't have a direct impact on the quality of the estimates.

A. Local minimality conditions for a integral functional

In light of the regressor form (5), it is instructive to search for the candidate switching time τ that minimizes the integral square error $J(\tau) = J_1(\tau) + J_2(\tau)$ with

$$\begin{aligned} J_1(\tau) &= \int_{t_0}^{t} |\hat{\theta}_1^T(\tau)\phi(s) - y(s)|^2 \,\mathrm{d}s \\ J_2(\tau) &= \int_{t}^{t_N} |\hat{\theta}_2^T(\tau)\phi(s) - y(s)|^2 \,\mathrm{d}s \end{aligned} \tag{6}$$

If we consider a single model switch, the parameter values that minimize the cost functionals in the subintervals $[t_0, t]$ and $[t, t_N]$ are, respectively,

$$\begin{aligned} \hat{\theta}_1(\tau) &= P(t_0, \tau) \int_{t_0}^{\tau} \phi(s) y^T(s) \,\mathrm{d}s \\ \hat{\theta}_2(\tau) &= P(\tau, t_N) \int_{\tau}^{t_N} \phi(s) y^T(s) \,\mathrm{d}s, \end{aligned} \tag{7}$$

where

$$P^{-1}(t_0, \tau) = \int_{t_0}^{\tau} \phi(s)\phi^T(s)\mathrm{d}s$$

and

$$P^{-1}(\tau, t_N) = \int_{\tau}^{t_N} \varphi(s)\varphi^T(s)\mathrm{d}s$$

were defined in the assumption that the inverses exist, a reasonable simplification provided no algorithms are made to depend on it.

The quantity $J(\tau)$ is continuously differentiable at times τ such that y is continuous. However 2nd order optimization methods will not be useful for finding τ, as the 2nd derivative of $J(\tau)$ depends on derivatives of the measured signal, not in general an available or computable quantity. As will be seen in the sequel, approximate 2nd order methods would not be effective either, because of the nonconvex nature of the cost functions. To understand the potential effectiveness of alternative search methods, we need to think about possible scenarios for the data–generating process itself.

B. Signals generated by a process that switches once

We now consider that $y(t) = \theta_\sigma^T \phi(t) + w(t)$, with the parameters θ_σ switching once at instant s_1 such that $t_0 \leq s_1 \leq t_N$:

$$\theta_\sigma = \begin{cases} \theta_1, & t \in [t_0, s_1) \\ \theta_2, & t \in [s_1, t_N]. \end{cases}$$

This case reveals enough about the qualitative behavior of the cost functional without distracting and unnecessary complications.

Further assume that $w(t) = 0$ and that $\phi(t)$ comprises high–frequency stationary terms, so that

$$\int_{t_a}^{t_b} \phi(t)\phi^T(t)\mathrm{d}t \approx \Phi(t_a - t_b)$$

for some constant matrix Φ, whenever the interval $t_b - t_a$ is not too small. This approximation would be absurd for unstable processes, or also for processes whose behavior changes substantially at switching times, however the idea of a stationary or ergodic regressor is a reasonable starting point for investigating the behavior of a system operating around a steady state.

Consider 1st the case when $\tau \in [t_0, s_1)$. Then

$$\begin{aligned} J(\tau) = \int_{t_0}^{\tau} &|(\hat\theta_1(\tau) - \theta_1)^T \phi(s)|^2 \,\mathrm{d}s \\ &+ \int_{\tau}^{s_1} |(\hat\theta_2(\tau) - \theta_1)^T \phi(s)|^2 \,\mathrm{d}s \\ &\quad + \int_{s_1}^{t_N} |(\hat\theta_2(\tau) - \theta_2)^T \phi(s)|^2 \,\mathrm{d}s \end{aligned} \quad (8)$$

where

$$\begin{aligned} \hat\theta_1(\tau) &= P(t_0, \tau) \int_{t_0}^{\tau} \phi(s)\phi^T(s)\,\mathrm{d}s\,\theta_1 = \theta_1 \\ \hat\theta_2(\tau) &= P(\tau, t_N) \left(\int_{\tau}^{s_1} \phi\phi^T(s)\,\mathrm{d}s\,\theta_1 + \int_{s_1}^{t_N} \phi\phi^T(s)\,\mathrm{d}s\,\theta_2 \right) \\ &= \frac{(s_1 - \tau)\theta_1 + (t_N - s_1)\theta_2}{t_N - \tau}, \end{aligned}$$

from which

$$\begin{aligned} \hat\theta_2(\tau) - \theta_1 &= \frac{t_N - s_1}{t_N - \tau}(-\theta_1 + \theta_2) \\ \hat\theta_2(\tau) - \theta_2 &= \frac{s_1 - \tau}{t_N - \tau}(\theta_1 - \theta_2). \end{aligned}$$

Writing $\Phi_{12} = \mathrm{tr}[(\theta_2 - \theta_1)^T \Phi(\theta_2 - \theta_1)]$,

$$\begin{aligned} J(\tau) &= \left((s_1 - \tau)\left(\tfrac{t_N - s_1}{t_N - \tau}\right)^2 + (t_N - s_1)\left(\tfrac{s_1 - \tau}{t_N - \tau}\right)^2 \right)\Phi_{12} \\ &= (t_N - s_1)\frac{s_1 - \tau}{t_N - \tau}\Phi_{12} \text{ for } \tau \in [t_0, s_1). \end{aligned} \quad (9)$$

Likewise when $\tau \in [s_1, t_N)$

$$\begin{aligned} J(\tau) = \int_{t_0}^{s_1} &|(\hat\theta_1(\tau) - \theta_1)^T \phi(s)|^2 \,\mathrm{d}s \\ &+ \int_{s_1}^{\tau} |(\hat\theta_1(\tau) - \theta_1)^T \phi(s)|^2 \mathrm{d}s \\ &\quad + \int_{\tau}^{t_N} |(\hat\theta_2(\tau) - \theta_2)^T \phi(s)|^2 \,\mathrm{d}s \end{aligned}$$

where

$$\begin{aligned} \hat\theta_1(\tau) - \theta_1 &= \frac{\tau - s_1}{\tau - t_0}(-\theta_1 + \theta_2) \\ \hat\theta_2(\tau) - \theta_2 &= \frac{s_1 - t_0}{\tau - t_0}(\theta_1 - \theta_2). \end{aligned}$$

Consequently

$$J(\tau) = (s_1 - t_0)\frac{\tau - s_1}{\tau - t_0}\Phi_{12} \text{ , for } \tau \in [s_1, t_N). \quad (10)$$

For $\tau = s_1$, the quantities in (8) and (9) coincide. An illustrative plot is in Figure 1. The plot resembles an Argentinian medialuna cut in two pieces and reassembled. It suggests that garden–variety optimization algorithms could be used to search for model switching instant, provided we don't get too technically demanding on the differentiability properties of the objective function. Notice further that the derivative of the quadratic integral cost is available in principle because the cost is defined as an integral, so the choice between a gradient–type or a derivative–free optimization method will come to the questions of taste and computation time.

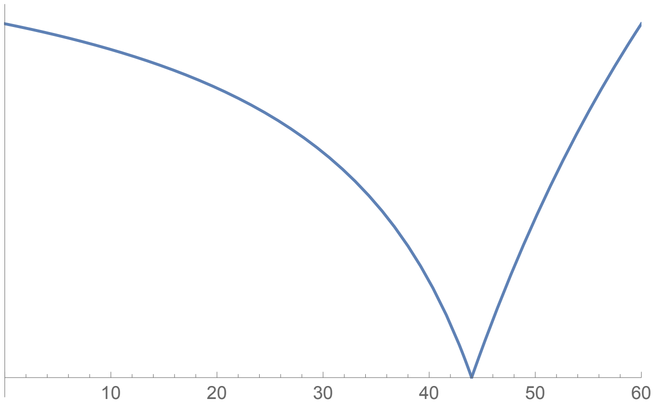

Fig. 1. Cost functional for once–switching system as function of model switching time.

C. Signals generated by a process that switches twice

What happens if we misunderestimate the number of switchings? If the model parameters $\hat\theta$ switch once, but the plant parameters θ switch twice, the fairly tedious calculations in the appendix give the formula for $J(\tau)$ found in equation (11), where we wrote

$$\begin{aligned} \Phi_{123} &= \mathrm{tr}\left[(\theta_2 - \theta_1)^T \Phi(\theta_3 - \theta_2)\right] \\ \Phi_{23} &= \mathrm{tr}\left[(\theta_3 - \theta_2)^T \Phi(\theta_3 - \theta_2)\right], \end{aligned}$$

and Φ_{12} as previously.

A sketch of the behavior of the error when θ switches twice but $\hat\theta$ only switches once is shown in Figure 2 for illustrative purposes. It shows that the minima are much less sharply defined than in the case when only one switching of the data–generating process occurs, suggesting that searches for individual switching instants are still feasible, but searching for all of them simultaneously, if we have a good idea about the number of switches, may be preferable.

It is also possible that we misoverestimate the number of switchings. Consider for example the case when the process switches once, as in Section III-B, but the model switches twice. Then formulas (9) and (10) still describe the residual, if we replace t_0 and t_N by subsequent model switching instants τ_a and τ_b such that $s_1 \in [\tau_a, \tau_b]$. It is in principle feasible to repeat these calculations for processes that switch several times, and with several model switching instants, however we have not found the messy computations to be particularly useful in picking optimization algorithms.

The computations given here are only very rough sketches. The point is to show that the cost function is clearly not convex, a fact that may contribute to the difficulty of simultaneously estimating parameters and switching times. However, minimizing the function $J(\tau)$ alone is feasible. As a proof of concept, in this paper we approach the problem with some very simple methods.

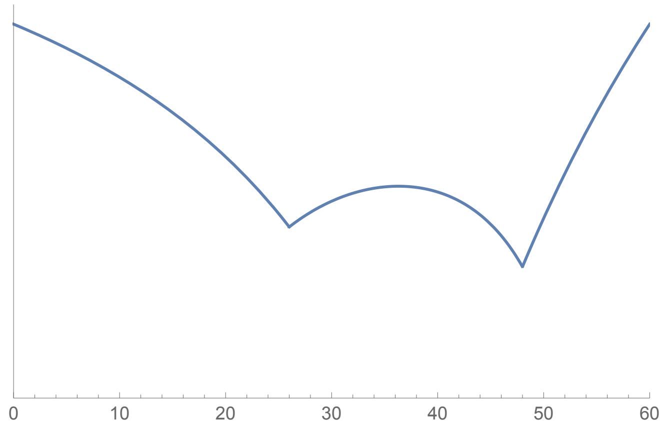

Fig. 2. Cost functional for twice–switching system as function of model switching time.

IV. Switching Time Estimation

With the construction presented in Section II, parameter estimation can be approached using least–squares or other proven methods, once model switching times have been assigned. In light of the approximate analysis in Section III, we can approach the assignment of switching times as an optimization problem over the total residual $J(\tau)$. The cost to be minimized is not convex, which means that Newton–like methods might not converge, even if 2nd derivatives were available. Gradient–based or descent methods may be used, but they also are not at their best behavior in nonconvex optimization. Sequential search algorithms are a possibility, particularly those ones that exploit ideas of dynamic programming (e.g., [4]). This approach was recently applied to ARX model segmentation [8] and achieves the exact global solution.

For situations in which a huge amount of data has to be handled and the computation burden is an issue, the bisection method [1, Ch.2] is proposed here as an alternative. Since the bisection algorithm is restricted to detecting a single changepoint, we resort to binary segmentation to address the multiple switching case. Binary segmentation starts by applying the bisection method to the entire dataset, to estimate the switching time τ^* that minimizes $J(\tau) = J_1(\tau) + J_2(\tau)$, where the continuous–time expressions in (6) are replaced with

$$J_1(\tau) = \sum_{k=t_0}^{\tau} |\hat{\theta}_1^T(\tau)\phi(t_k) - y(t_k)|^2$$
$$J_2(\tau) = \sum_{k=\tau}^{t_N} |\hat{\theta}_2^T(\tau)\phi(t_k) - y(t_k)|^2.$$

The least–squares estimates $\hat{\theta}_1$ and $\hat{\theta}_2$ are computed based on regression model (5) evaluated at discrete time instants $t = t_k$, for data segments $k = \{1, \ldots, \tau\}$ and $k = \{\tau, \ldots, N\}$, respectively.

Next, the obtained changepoint is evaluated according to

$$J_1(\tau^*) + J_2(\tau^*) < \beta^{-1}\breve{J}, \tag{12}$$

where

$$\breve{J} = \sum_{k=1}^{t_N} |\hat{\theta}_1^T(\tau)\phi(t_k) - y(t_k)|^2$$

is the cost function achieved by assuming that there is no changepoint (so a single segment is considered and $\tau = t_N$), and $\beta > 1$ is a penalty term used to avoid overfitting (the higher β, the higher is the improvement necessary to admit a model switch). If τ^* does not met inequality (12), it is not assigned as a switching–point. If it does, the data are split into two new segments: a sequence up to τ^*, and another from this point on. Then bisection method is applied to each new partition. The procedure is repeated until either no switching points are detected or the number of changes exceeds a pre–specified threshold.

Although the bisection method is not guaranteed to find the global minimum of $J(\tau)$, it is simple, computationally efficient, and, as illustrated in the next section, effective in finding reasonable estimates of the switching time, provided few changes occur. In fact, by extrapolating the results in Sections III-B and III-C, as the number of switchings increases, the minimum of $J(\theta) = J_1(\tau^*) + J_2(\tau^*)$ becomes less pronounced. Thus, if the system switches several times, the first iterations of the binary segmentation algorithm provide minor improvements due to a single model change,

$$J(\tau) = \begin{cases} \frac{(s_1-\tau)(t_N-s_1)}{t_N-\tau}\Phi_{12} + 2\frac{(s_1-\tau)(t_N-s_2)}{t_N-\tau}\Phi_{123} + \frac{(t_N-s_2)(s_2-\tau)}{t_N-\tau}\Phi_{23}, \text{ for } \tau \in [t_0, s_1) \\ \frac{(s_1-t_0)(\tau-s_1)}{\tau-t_0}\Phi_{12} + \frac{(t_N-s_2)(s_2-\tau)}{t_N-\tau}\Phi_{32}, \text{ for } \tau \in [s_1, s_2) \\ \frac{(s_1-t_0)(\tau-s_1)}{\tau-t_0}\Phi_{12} + 2\frac{(s_1-t_0)(\tau-s_2)}{\tau-t_0}\Phi_{123} + \frac{(\tau-s_2)(s_2-t_0)}{\tau-t_0}\Phi_{23}, \text{ for } \tau \in [s_2, t_N]. \end{cases} \tag{11}$$

so that a β very close to one is required to detect the changepoint. However, setting $\beta \approx 1$ can lead to the detection of false switching instants. To deal with this behavior, we employed a varying penalty term β^r which increases at each binary segmentation iteration r. This strategy is easily implemented by replacing β^{-1} with β^{-r} on the right-hand side of inequality (12).

V. Simulations

To evaluate the binary segmentation strategy, two simulation examples from [7] are employed. The results are compared with the sum–of–norms regularization method based on convex programming presented in the same work.

Example 1: Consider a discrete–time switched system

$$\begin{aligned} y_0(k) &= a_1 y_0(k-1) - 0.7 y_0(k-2) + u(k-1) \\ y(k) &= y_0(k) + w(k), \end{aligned}$$

which is excited using $u(k) \sim \mathcal{N}(0,1)$ of length $N = 5000$. The noise–free output y_0 is is corrupted by zero–mean white noise $w(k)$, whose variance is adjusted to provide a signal–to–noise ratio (SNR) of 15dB. A Monte Carlo simulation with 50 runs is performed. In each one, the parameter a_1 switches from 1.5 to 1.3 at a random instant between $[15, 4985]$. Although the analysis in the previous sections has been carried out in continuous time, the tests show that the qualitative results apply to the sampled–data case.

The penalty term of the binary segmentation algorithm of Section IV is set to $\beta = 1.1$ (which means an improvement of 10% in the first iteration, 21% in the second, and so on). The error histogram of the estimate $\hat{\tau}$ and the one achieved using the convex programming approach in [7] are presented in Figure 3. The proposed approach estimated accurately the switching times over the simulations and exhibited a lower variability.

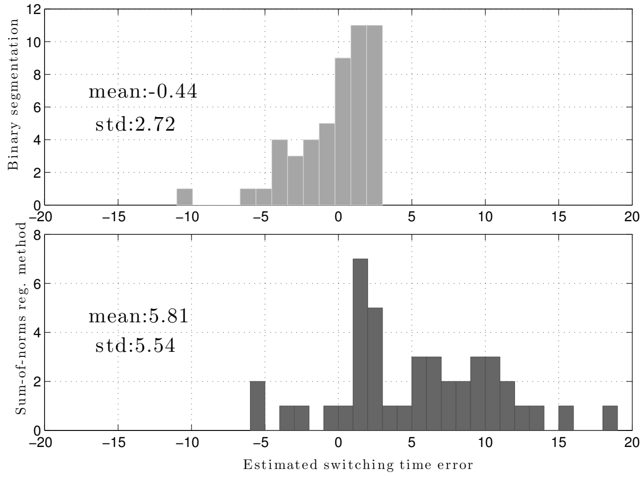

Fig. 3. Histograms of the switching instant estimation error of Example 1.

Example 2: Now consider the discrete–time system

$$\begin{aligned} y_0(k) &= a_1 y_0(k-1) - 0.7 y(k-2) \\ &\quad + u(k-1) + 0.5u(k-2) \\ y(k) &= y_0(k) + w(k), \end{aligned}$$

where $u(k) \sim \mathcal{N}(0,1)$ of length $N = 2000$. The coefficient a_1 switches from 1.5 to 1.3 at $k = 400$ and returns to 1.5 at $k = 1500$. As in the previous example, $w(k)$ is zero–mean white noise with an SNR of 15dB. The changepoint detection penalty term of the binary segmentation algorithm is set to $\beta = 1.04$. Figure 4 shows the time-varying parameter estimates for 50 runs of a Monte Carlo simulation. For our method a_1 is recovered by converting the estimated model for each segment to the discrete–time domain. Thanks to the filtering inherent to the parameterization described in Section II, the estimates provided by our approach presents a lower bias. In addition, the estimates of the change times are more closely spaced together among the considered realizations, as depicted in Figure 5 that shows the switching parameter estimate at the vicinities of the switching times. The latter property is arguably more important, because after segmentation is performed, more sophisticated LTI identification algorithms can be applied to the partitioned data to get more accurate models for each segment.

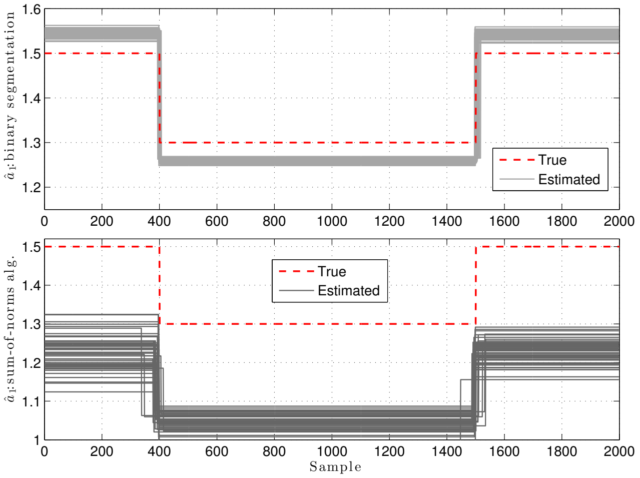

Fig. 4. Switching parameter estimates.

The mean time considering the 50 runs of the binary segmentation algorithm was 0.048s, while the convex programming method[2] from [7] spent 70.7s in an iMac with 2.7GHz Intel Core i5 processor and 16GB memory. Even though a direct comparison may be unfair (the methods have distinct proposals: while binary segmentation searches for a restricted amount of changes per iteration, in the other method the computational complexity does not depend on the number of segments), such discrepancy together with the previous results highlight the advantage of tackling the

[2] To improve the estimate, two iterations of the refinement algorithm discussed in [7] is applied.

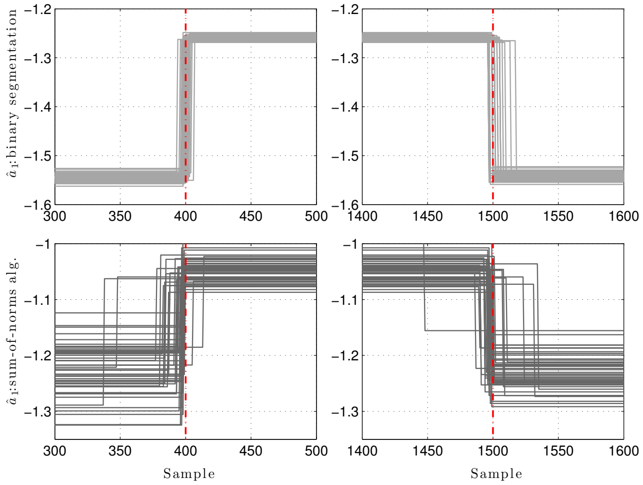

Fig. 5. Switching parameter estimates at the vicinities of the switching times.

parameter estimation and the data segmentation problems separately, at least for the case of infrequent switches.

VI. Conclusion

The overall optimization problem of data segmentation and parameter estimation for linear models can be computationally intractable, if mixed discrete–continuous optimization problems with nonconvex objectives appear. Our approach is to separate it into more manageable tasks. Once a suitable linear system structure is chosen, parameter estimation may be approached by elementary methods. Here we chose the least–squares approach, which leads to closed–form solutions, given the switching times.

Switching time estimation leads to nonconvex optimization. In general a nonconvex problem can be very difficult, which may be why the joint estimation problem is so complicated. However, estimating the switching times by themselves is not so hard. The search is essentially one–dimensional; in the case of multiple switching times, it is a set of one–dimensional searches, which are feasible despite lack of convexity. As a proof of concept, we tested binary search methods together with simple heuristics. In simple problems, the tests indicate that the methods converge orders of magnitude faster than other solutions proposed in the literature, obtaining comparable or better performance. We expect the advantages will become more pronounced for more complex problems, with longer data sets, and for higher–order and multivariable models.

Work on improved switching time estimation algorithms is under way. We intend to explore the use of more advanced optimization methods, such as the barycenter method for direct optimization [10] or perhaps other derivative–free optimization techniques. See also [11], where the barycenter method was employed successfully to deal with the issue of optimizing filter poles in linear system identificaion. Notice that convex optimization methods will not be effective; this is the main lesson from the digression presented in Section III! At this point in time, the naive bisection method employed as a proof of concept is sufficient to establish the performance improvements we wished to obtain with our approach.

After the models and switching times are obtained, it may be useful to classify the models into a number smaller than that of the time intervals. The main question is determining whether models of non–adjacent segments could be classified as the same model. The present paper does not consider this issue, but we may say that splitting the problem into subtasks makes it more tractable computationally.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support of this research by Instituto Mauá de Tecnologia (IMT). We thank Henrik Ohlsson for kindly making available an implementation of the convex programming method.

References

[1] M. Bartholomew-Biggs, *Nonlinear Optimization with Financial Applications*. Springer, 2005.

[2] P. Brunovský, “A classification of linear controllable systems,” *Kybernetika*, vol. 6, pp. 173–188, 1970.

[3] 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.

[4] B. Jackson, J. D. Scargle, D. Barnes, S. Arabhi, A. Alt, P. Gioumousis, E. Gwin, P. Sangtrakulcharoen, L. Tan, and T. T. Tsai, “An algorithm for optimal partitioning of data on an interval,” *IEEE Signal Processing Letters*, vol. 12, no. 2, pp. 105–108, Feb 2005.

[5] D. G. Luenberger, “Canonical forms for linear multivariable systems,” *IEEE Transactions on Automatic Control*, vol. AC-12, pp. 290–293, 1967.

[6] 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.

[7] H. Ohlsson, L. Ljung, and S. Boyd, “Segmentation of ARX-models using sum-of-norms regularization,” *Automatica*, vol. 46, no. 6, pp. 1107–1111, June 2010.

[8] N. Ozay, “An exact and efficient algorithm for segmentation of ARX models,” in *2016 American Control Conference (ACC)*, Boston, MA, USA, July 2016, pp. 38–41.

[9] N. Ozay, M. Sznaier, C. M. Lagoa, and O. I. Camps, “A sparsification approach to set membership identification of switched affine systems,” *IEEE Transactions on Automatic Control*, vol. 57, no. 3, pp. 634–648, March 2012.

[10] F. M. Pait, “The Barycenter Method for Direct Optimization,” *ArXiv e-prints*, Jan. 2018. [Online]. Available: https://arxiv.org/abs/1801.10533

[11] 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.

[12] R. A. Romano, F. Pait, and P. L. dos Santos, “Obtaining multivariable continuous–time models from sampled data,” in *2017 American Control Conference (ACC)*, Seattle, 2017.

Appendix

Consider the case when the model parameters $\hat{\theta}$ switch once, but the plants parameters θ switch twice, i.e.

$$\theta_\sigma = \begin{cases} \theta_1, & t \in [t_0, s_1) \\ \theta_2, & t \in [s_1, s_2], \\ \theta_3, & t \in [s_2, t_N]. \end{cases} \tag{13}$$

There are 3 cases to consider, according to the instant when model parameters switch.

1) For $\tau \in [t_0, s_1)$

$$\begin{aligned} J(\tau) = \int_{t_0}^{\tau} |(\hat{\theta}_1(\tau) - \theta_1)^T \phi(s)|^2 \,\mathrm{d}s + \int_{\tau}^{s_1} |(\hat{\theta}_2(\tau) - \theta_1)^T \phi(s)|^2 \,\mathrm{d}s \\ + \int_{s_1}^{s_2} |(\hat{\theta}_2(\tau) - \theta_2)^T \phi(s)|^2 \,\mathrm{d}s + \int_{s_2}^{t_N} |(\hat{\theta}_2(\tau) - \theta_3)^T \phi(s)|^2 \,\mathrm{d}s \end{aligned}$$

with

$$\begin{aligned} \hat{\theta}_1(\tau) &= P(t_0, \tau) \int_{t_0}^{\tau} \phi(s)\phi^T(s) \,\mathrm{d}s \,\theta_1 = \theta_1 \\ \hat{\theta}_2(\tau) &= P(\tau, t_N) \left(\int_{\tau}^{s_1} \phi(s)\phi^T(s) \,\mathrm{d}s \,\theta_1 + \int_{s_1}^{s_2} \phi(s)\phi^T(s) \,\mathrm{d}s \,\theta_2 + \int_{s_2}^{t_N} \phi(s)\phi^T(s) \,\mathrm{d}s \,\theta_3 \right) \\ &\approx \frac{(s_1 - \tau)\theta_1 + (s_2 - s_1)\theta_2 + (t_N - s_2)\theta_3}{t_N - \tau}. \end{aligned}$$

The approximations in Section III-B will be used throughout. Hence

$$\begin{aligned} \hat{\theta}_1(\tau) - \theta_1 &= 0 \\ \hat{\theta}_2(\tau) - \theta_1 &= \frac{(t_N - s_1)(\theta_2 - \theta_1) + (t_N - s_2)(\theta_3 - \theta_2)}{t_N - \tau} \\ \hat{\theta}_2(\tau) - \theta_2 &= \frac{(\tau - s_1)(\theta_2 - \theta_1) + (t_N - s_2)(\theta_3 - \theta_2)}{t_N - \tau} \\ \hat{\theta}_2(\tau) - \theta_3 &= \frac{(\tau - s_1)(\theta_2 - \theta_1) + (\tau - s_2)(\theta_3 - \theta_2)}{t_N - \tau} \end{aligned}$$

and

$$\begin{aligned} J(\tau) = & \frac{(s_1 - \tau)(t_N - s_1)^2 + (s_2 - s_1)(\tau - s_1)^2 + (t_N - s_2)(\tau - s_1)^2}{(t_N - \tau)^2} \operatorname{tr}[(\theta_2 - \theta_1)^T \Phi (\theta_2 - \theta_1)] \\ & + 2 \frac{(s_1 - \tau)(t_N - s_1)(t_N - s_2) + (s_2 - s_1)(\tau - s_1)(t_N - s_2) + (t_N - s_2)(\tau - s_1)(\tau - s_2)}{(t_N - \tau)^2} \operatorname{tr}[(\theta_2 - \theta_1)^T \Phi (\theta_3 - \theta_2)] \\ & + \frac{(s_1 - \tau)(t_N - s_2)^2 + (s_2 - s_1)(t_N - s_2)^2 + (t_N - s_2)(\tau - s_2)^2}{(t_N - \tau)^2} \operatorname{tr}[(\theta_3 - \theta_2)^T \Phi (\theta_3 - \theta_2)] \\ & = \frac{(s_1 - \tau)(t_N - s_1)}{t_N - \tau} \Phi_{12} + 2 \frac{(s_1 - \tau)(t_N - s_2)}{t_N - \tau} \Phi_{123} + \frac{(t_N - s_2)(s_2 - \tau)}{t_N - \tau} \Phi_{23}. \end{aligned} \tag{14}$$

2) $\tau \in [s_1, s_2)$

$$\begin{aligned} J(\tau) = \int_{t_0}^{s_1} |(\hat{\theta}_1(\tau) - \theta_1)^T \phi(s)|^2 \,\mathrm{d}s + \int_{s_1}^{\tau} |(\hat{\theta}_1(\tau) - \theta_2)^T \phi(s)|^2 \,\mathrm{d}s \\ + \int_{\tau}^{s_2} |(\hat{\theta}_2(\tau) - \theta_2)^T \phi(s)|^2 \,\mathrm{d}s + \int_{s_2}^{t_N} |(\hat{\theta}_2(\tau) - \theta_3)^T \phi(s)|^2 \,\mathrm{d}s \end{aligned} \tag{15}$$

with

$$\begin{aligned} \hat{\theta}_1(\tau) &= P(t_0, \tau) \left(\int_{t_0}^{s_1} \phi(s)\phi^T(s) \,\mathrm{d}s \,\theta_1 + \int_{s_1}^{\tau} \phi(s)\phi^T(s) \,\mathrm{d}s \,\theta_2 \right) \approx \frac{(s_1 - t_0)\theta_1 + (\tau - s_1)\theta_2}{\tau - t_0}, \\ \hat{\theta}_2(\tau) &= P(\tau, t_N) \left(\int_{\tau}^{s_2} \phi(s)\phi^T(s) \,\mathrm{d}s \,\theta_2 + \int_{s_2}^{t_N} \phi(s)\phi^T(s) \,\mathrm{d}s \,\theta_3 \right) \approx \frac{(s_2 - \tau)\theta_2 + (t_N - s_2)\theta_3}{t_N - \tau}. \end{aligned}$$

Hence

$$
\begin{aligned}
\hat{\theta}_1(\tau) - \theta_1 &= \frac{\tau - s_1}{\tau - t_0}(\theta_2 - \theta_1) \\
\hat{\theta}_1(\tau) - \theta_2 &= \frac{t_0 - s_1}{\tau - t_0}(\theta_2 - \theta_1) \\
\hat{\theta}_2(\tau) - \theta_2 &= \frac{t_N - s_2}{t_N - \tau}(\theta_3 - \theta_2) \\
\hat{\theta}_2(\tau) - \theta_3 &= \frac{\tau - s_2}{t_N - \tau}(\theta_3 - \theta_2)
\end{aligned}
$$

and

$$
\begin{aligned}
J(\tau) = {} & \left((s_1 - t_0)\left(\frac{\tau - s_1}{\tau - t_0}\right)^2 + (\tau - s_1)\left(\frac{t_0 - s_1}{\tau - t_0}\right)^2 \right) \mathrm{tr}[(\theta_2 - \theta_1)^T \Phi (\theta_2 - \theta_1)] \\
& + \left((s_2 - \tau)\left(\frac{t_N - s_2}{t_N - \tau}\right)^2 + (t_N - s_2)\left(\frac{\tau - s_2}{t_N - \tau}\right)^2 \right) \mathrm{tr}[(\theta_3 - \theta_2)^T \Phi (\theta_3 - \theta_2)] \\
& = \frac{(s_1 - t_0)(\tau - s_1)}{\tau - t_0}\Phi_{12} + \frac{(t_N - s_2)(s_2 - \tau)}{t_N - \tau}\Phi_{23}. \qquad (16)
\end{aligned}
$$

3) $\tau \in [s_2, t_N]$

$$
\begin{aligned}
J(\tau) = {} & \int_{t_0}^{s_1} |(\hat{\theta}_1(\tau) - \theta_1)^T \phi(s)|^2 \, \mathrm{d}s + \int_{s_1}^{s_2} |(\hat{\theta}_1(\tau) - \theta_2)^T \phi(s)|^2 \, \mathrm{d}s \\
& + \int_{s_2}^{\tau} |(\hat{\theta}_1(\tau) - \theta_3)^T \phi(s)|^2 \, \mathrm{d}s + \int_{\tau}^{t_N} |(\hat{\theta}_2(\tau) - \theta_3)^T \phi(s)|^2 \, \mathrm{d}s
\end{aligned}
$$

with

$$
\begin{aligned}
\hat{\theta}_1(\tau) &= P(t_0, \tau) \left(\int_{t_0}^{s_1} \phi(s)\phi^T(s) \, \mathrm{d}s \, \theta_1 + \int_{s_1}^{s_2} \phi(s)\phi^T(s) \, \mathrm{d}s \, \theta_2 + \int_{s_2}^{\tau} \phi(s)\phi^T(s) \, \mathrm{d}s \, \theta_3 \right) \\
&\approx \frac{(s_1 - t_0)\theta_1 + (s_2 - s_1)\theta_2 + (\tau - s_2)\theta_3}{\tau - t_0} \\
\hat{\theta}_2(\tau) &= P(\tau, t_N) \int_{\tau}^{t_N} \phi(s)\phi^T(s) \, \mathrm{d}s \, \theta_3 = \theta_3.
\end{aligned}
$$

Again using the same approximations,

$$
\begin{aligned}
\hat{\theta}_1(\tau) - \theta_1 &= \frac{(\tau - s_1)(\theta_2 - \theta_1) + (\tau - s_2)(\theta_3 - \theta_2)}{\tau - t_0} \\
\hat{\theta}_1(\tau) - \theta_2 &= \frac{(t_0 - s_1)(\theta_2 - \theta_1) + (\tau - s_2)(\theta_3 - \theta_2)}{\tau - t_0} \\
\hat{\theta}_1(\tau) - \theta_3 &= \frac{(t_0 - s_1)(\theta_2 - \theta_1) + (t_0 - s_2)(\theta_3 - \theta_2)}{\tau - t_0} \\
\hat{\theta}_2(\tau) - \theta_3 &= 0
\end{aligned}
$$

and

$$
\begin{aligned}
J(\tau) = {} & \frac{(s_1 - t_0)(\tau - s_1)^2 + (s_2 - s_1)(t_0 - s_1)^2 + (\tau - s_2)(t_0 - s_1)^2}{(\tau - t_0)^2} \mathrm{tr}[(\theta_2 - \theta_1)^T \Phi (\theta_2 - \theta_1)] \\
& + 2\frac{(s_1 - t_0)(\tau - s_1)(\tau - s_2) + (s_2 - s_1)(t_0 - s_1)(\tau - s_2) + (\tau - s_2)(t_0 - s_1)(t_0 - s_2)}{(\tau - t_0)^2} \mathrm{tr}[(\theta_2 - \theta_1)^T \Phi (\theta_3 - \theta_2)] \\
& + \frac{(s_1 - t_0)(\tau - s_2)^2 + (s_2 - s_1)(\tau - s_2)^2 + (\tau - s_2)(t_0 - s_2)^2}{(\tau - t_0)^2} \mathrm{tr}[(\theta_3 - \theta_2)^T \Phi (\theta_3 - \theta_2)] \\
& = \frac{(s_1 - t_0)(\tau - s_1)}{\tau - t_0}\Phi_{12} + 2\frac{(s_1 - t_0)(\tau - s_2)}{\tau - t_0}\Phi_{123} + \frac{(\tau - s_2)(s_2 - t_0)}{\tau - t_0} + \Phi_{23}. \qquad (17)
\end{aligned}
$$

Equations (14), (16), and (17) compose the formula in Section III-C.