An exact and efficient algorithm for segmentation of ARX models

Necmiye Ozay

***Abstract*— We consider the problem of segmentation of autoregressive models with exogenous inputs (ARX models). This problem, where the goal is to determine the parameters of a sequence of ARX models that can explain a given input/output data within some noise bound, has attracted considerable attention in recent years. Most of the recently proposed approaches are based on convex relaxations. Although efficient, these approaches do not necessarily lead to optimal solutions. In the present paper, by exploiting some early results in dynamic programming, we show that an optimal solution can indeed be obtained in polynomial-time. One salient feature of the proposed approach is that exploration of the model complexity/quality of the fit trade-off space comes with negligible additional computational cost. We discuss several other properties of the proposed approach and compare it with existing approaches on numerical examples, which show that the proposed approach is consistently faster.**

I. INTRODUCTION AND MOTIVATION

The problem of segmentation of ARX models has attracted considerable attention in recent years. There are several applications of this problem in anomaly and change detection both in dynamical systems [9], [7], [10] and in computer vision [8]. This problem has been shown to be equivalent to switched system identification with minimum number of switches [9], [10]. Segmentation problems are usually combinatorial, however, by recasting the problem into a sparsification form, it is possible to leverage ideas from compressed sensing literature to efficiently solve this problem [9], [7], [10]. Moreover, as shown in [9], [10], for ARX models with ℓ_∞-norm bounded noise, a greedy algorithm gives an exact solution. For the case of ℓ_2-norm bounded noise, in addition to earlier ℓ_1-norm relaxation proposed in [9], [10], various other relaxation techniques have been proposed recently [7], [11]. Although these relaxations lead to convex optimization problems, it is not difficult to construct examples where the solution of the relaxed problem is far from the solution of the original problem. Also, one usually needs to adjust a regularization parameter to trade-off between the complexity of the model (i.e., the number of segments) and the quality of the fit.

In this paper, we propose a polynomial-time yet exact algorithm for segmentation of ARX models by exploiting ideas from early days of dynamic programming [2]. Similar ideas have been adopted in the econometrics [1] and signal processing literatures [3], [4] for certain change detection problems. Our main contribution is to adapt these ideas to the ARX model segmentation problem and bring this powerful tool to the attention of the community working on this problem. In particular, we pose the problem in a fairly general form and present conditions under which this general form can be solved using dynamic programing. It turns out that different variants of the ARX model segmentation problem considered by the controls community satisfy these conditions, therefore amenable to exact solutions. Compared to the existing techniques for ARX model segmentation, the proposed algorithm not only has stronger optimality guarantees but also shown to be computationally more efficient both in theory and in practice as demonstrated by various numerical examples. Another salient feature of the algorithm is that exploration of the whole complexity-quality of the fit trade-off space does not change the complexity significantly and can still be done in polynomial-time.

Notation: $\|\mathbf{x}\|_0$ denotes the ℓ_0-quasinorm (ℓ_0-norm, for short) that is equal to number of non-zero entries of the vector $\mathbf{x}$.

II. PROBLEM SETUP

In this paper, we consider time-varying affine autoregressive exogenous models of the form:

$$y_t = \sum_{i=1}^{n_a} a_t^i y_{t-i} + \sum_{i=1}^{n_c} c_t^i u_{t-i} + k_t + \eta_t \tag{1}$$

where u, y and η denote the input, output and noise, respectively, and where $t \in [t_o, N]$, with $t_o = \max(n_a, n_c)$. The parameters $\mathbf{p}_t \doteq [a_t^1, \ldots, a_t^{n_a}, c_t^1, \ldots, c_t^{n_c}, k_t]'$ are unknown. When the parameter vector is constant (i.e., $\mathbf{p}_t = \mathbf{p}^*$ for all t), we recover the time-invariant ARX models [5]. For notational simplicity, let the regressor vectors be defined as $\mathbf{r}_t \doteq [y_{t-1}, \ldots, y_{t-n_a}, u_{t-1}, \ldots, u_{t-n_c}, 1]'$ for $t \in [t_o, N]$. Then, the model (1) can be written as

$$y_t = \mathbf{p}_t' \mathbf{r}_t + \eta_t. \tag{2}$$

Given input/output data over a time horizon, the goal of segmentation of ARX models is to find a "simple" model of the form (1), where simplicity is usually measured by the number of changes in the parameter vector, that provides a "good" representation of data. This problem has been formalized in several different ways in the literature.

Let $f : \mathbb{R}^{N-t_o+1} \to \mathbb{R}$ be the fitting error function, a non-negative function that measures the quality of the fit[1]. Given input/output data $\{u_t, y_t\}_{t=0}^N$ over an interval $[0, N]$, we consider following variants of the segmentation problem.

This work was supported by DARPA grant N66001-14-1-4045. The author is with the EECS Department, University of Michigan, Ann Arbor, MI 48109, email: necmiye@umich.edu.

[1]Note that N varies depending on the amount of available data. That is, f is a function with indefinite arity.

Problem 1: Minimum number of switches with bounded fitting error:

$$\begin{array}{ll} \min_{\mathbf{p}_{t_o:N}} & \|[||\mathbf{p}_{t_o+1}-\mathbf{p}_{t_o}||,\ldots,||\mathbf{p}_N-\mathbf{p}_{N-1}||]'\|_0 \\ \text{s.t} & f(y_{t_o}-\mathbf{p}'_{t_o}\mathbf{r}_{t_o},\ldots,y_N-\mathbf{p}'_N\mathbf{r}_N)\leq\epsilon_1. \end{array} \tag{3}$$

Problem 2: Minimum fitting error with bounded number of switches:

$$\begin{array}{ll} \min_{\mathbf{p}_{t_o:N}} & f(y_{t_o}-\mathbf{p}'_{t_o}\mathbf{r}_{t_o},\ldots,y_N-\mathbf{p}'_N\mathbf{r}_N) \\ \text{s.t} & \|[||\mathbf{p}_{t_o+1}-\mathbf{p}_{t_o}||,\ldots,||\mathbf{p}_N-\mathbf{p}_{N-1}||]'\|_0\leq\epsilon_2. \end{array} \tag{4}$$

Problem 3: Unconstrained regularized segmentation:

$$\begin{array}{ll} \min_{\mathbf{p}_{t_o:N}} & f(y_{t_o}-\mathbf{p}'_{t_o}\mathbf{r}_{t_o},\ldots,y_N-\mathbf{p}'_N\mathbf{r}_N)+ \\ & \lambda\|[||\mathbf{p}_{t_o+1}-\mathbf{p}_{t_o}||,\ldots,||\mathbf{p}_N-\mathbf{p}_{N-1}||]'\|_0. \end{array} \tag{5}$$

These problems are tightly related. That is, when the minimum of f is a nonincreasing function of the number of changes in the parameter vector, for each fixed value of one of the variables $\epsilon_1,\epsilon_2,\lambda$, one can find a value for the other two variables so that an optimizer of one problem is also an optimizer of the other problems.

III. DYNAMIC PROGRAMMING FORMULATION

In this section, we present a polynomial-time algorithm that solves all the three problems in section II. The main insight is that although the number of possible segmentations is exponential in the horizon N, there are polynomially many segments. In fact the number of segments is quadratic in N.

Let us define the following subproblem for $t_1\leq t_2$:

$$e([t_1,t_2])\doteq\min_{\mathbf{p}} f(y_{t_1}-\mathbf{p}'\mathbf{r}_{t_1},\ldots,y_{t_2}-\mathbf{p}'\mathbf{r}_{t_2}), \tag{6}$$

that is, $e([t_1,t_2])$ is the fitting error when the data $\{u_t,y_t\}_{t=t_1}^{t_2}$ is modeled by a time-invariant ARX model.

Before we proceed, we make the following assumptions:

A1. The value $e([t_1,t_2])$ in (6) can be computed efficiently (i.e., in polynomial-time).

A2. The fitting error function f has the following form:

$$f(\eta_{t_o},\eta_{t_1},\ldots,\eta_N)=\sum_{t=t_o}^{N}\bar{f}(\eta_t), \tag{7}$$

for some non-negative function $\bar{f}:\mathbb{R}\to\mathbb{R}_{\geq 0}$.

For a given data sequence $\{u_t,y_t\}_{t=0}^{T}$ with $T\geq t_0$ and an integer $m>1$, the minimum fitting error with $m-1$ changes in the parameter vector is given by:

$$\begin{array}{ll} E(m,T) & =\min_{\mathbf{p}_{t_o:T}} f(y_{t_o}-\mathbf{p}'_{t_o}\mathbf{r}_{t_o},\ldots,y_T-\mathbf{p}'_T\mathbf{r}_T) \\ \text{s.t} & \|[||\mathbf{p}_{t_o+1}-\mathbf{p}_{t_o}||,\ldots,||\mathbf{p}_T-\mathbf{p}_{T-1}||]'\|_0\leq m. \end{array} \tag{8}$$

Theorem 1: Under assumption A2, the optimal value of Problem 2 is $E(\lfloor\epsilon_2\rfloor,N)$ and can be recursively computed as follows:

$$E(0,T)=e([t_o,T])\ \text{ for all } T\in[t_o,N], \tag{9}$$

and for $m=1,\ldots,\lfloor\epsilon_2\rfloor$ and for $T=\max(m+1,t_0),\ldots,N$,

$$E(m,T)=\min_{m\leq j\leq T-1}\left[E(m-1,j)+e([j+1,T])\right]. \tag{10}$$

Proof: The optimal value being $E(\lfloor\epsilon_2\rfloor,N)$ is clear from the definition of $E(m,T)$ and the fact that ℓ_0-norm only takes integer values. The recursion in (10) follows from assumption A2 and the principle of optimality. That is, if $\bar{\mathbf{p}}_{t_o:T}$ is an optimizer achieving $E(m,T)$, then for $t'=\max\{t\in[t_o+1,T]\mid\bar{\mathbf{p}}_t\neq\bar{\mathbf{p}}_{t-1}\}$, $\bar{\mathbf{p}}_{t_o:t'}$ must be an optimizer achieving $E(m-1,t')$. Otherwise, there would exist $\tilde{\mathbf{p}}_{t_o:t'}$ with $\|[||\tilde{\mathbf{p}}_{t_o+1}-\tilde{\mathbf{p}}_{t_o}||,\ldots,||\tilde{\mathbf{p}}_{t'}-\tilde{\mathbf{p}}_{t'-1}||]'\|_0\leq m-1$ such that $\sum_{t=t_o}^{t'}\bar{f}(y_t-\tilde{\mathbf{p}}_t'\mathbf{r}_t)\leq\sum_{t=t_o}^{t'}\bar{f}(y_t-\bar{\mathbf{p}}_t'\mathbf{r}_t)$, which contradicts to the optimality of $\bar{\mathbf{p}}_{t_o:T}$. ■

Note that the optimizers can be obtained by keeping track of the optimizing indices in j^* in equation (10) and corresponding optimizer of $e([j^*+1,T])$ in (6).

Theorem 2: Under assumptions A1 and A2, Problems 1,2 and 3 can be solved in polynomial time.

Proof: Assume $e([t_1,t_2])$ is computed and stored for $t_o\leq t_1\leq t_2\leq N$. This requires computing $e([t_1,t_2])$ $O(N^2)$ times. Then, running the recursion in Theorem 1 until $m=m^*$ requires a total of $O(m^*N^2)$ additions and $O(m^*N^2)$ comparisons for finding the minima. Therefore, under assumption A1, Problem 2 can be solved in polynomial time.

Problem 1 and 3 can be solved with the same recursion. For Problem 1, consider starting with $m=0$ and incrementing m until $E(m,N)\leq\epsilon_1$. Similarly for Problem 3, consider keeping track of $E(m,N)+\lambda m$ value as m is incremented until N.[2] Therefore, these problems can also be solved in polynomial time. ■

One salient feature of the recursion in equation (10) is that once $E(m,N)$ for all m upto $N-1$ is computed, it is possible to explore all the model complexity-quality of the fit trade-off space by comparing $E(m,N)$ values for different m (or similarly by considering different λ values) in linear time, where the model complexity is measured by the number of segments and the quality of the fit is measured by the total fitting error.

A few remarks on assumptions A1 and A2 are in order. Assumption A1 holds true whenever the fitting error function f is convex in its arguments. Since composition of a convex function with an affine function preserves convexity, the optimization problem (6) is a convex programming problem in the unknown parameter $\mathbf{p}$ when f is convex. For instance, one common measure of the quality of the fit is via the norm of the error and norms are convex functions allowing problem (6) to be solved efficiently. On the other hand, assumption A2 is slightly stronger than what is required and it can be relaxed. Dynamic programming can be used for any error function f' whose composition with a monotone non-decreasing function is in the form (7) (e.g., multiplicative errors with log, ℓ_2-norm with square, etc.).

IV. EXAMPLES

In this section, we demonstrate the effectiveness of the proposed approach on some numerical examples from [7],

[2] Early termination is possible when $E(m,N)$ drops below a certain value.

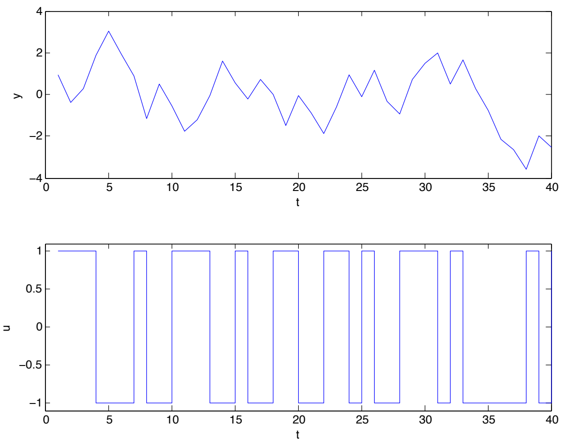

Fig. 1. Data for Example 1.

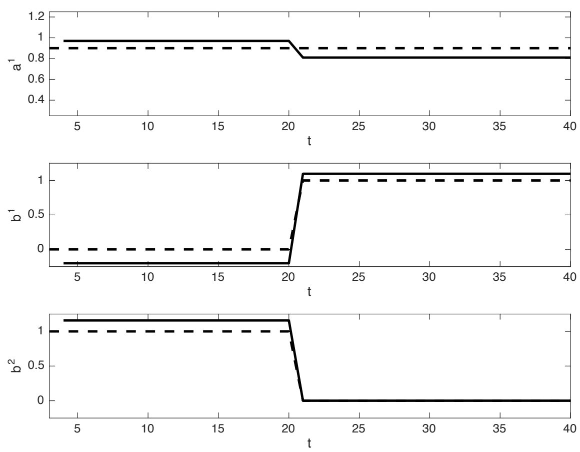

Fig. 2. Segmentation results, estimated parameter values (solid), and true parameter values (dashed) for Example 1 when solved using Problem 2 with $\epsilon_2 = 2$.

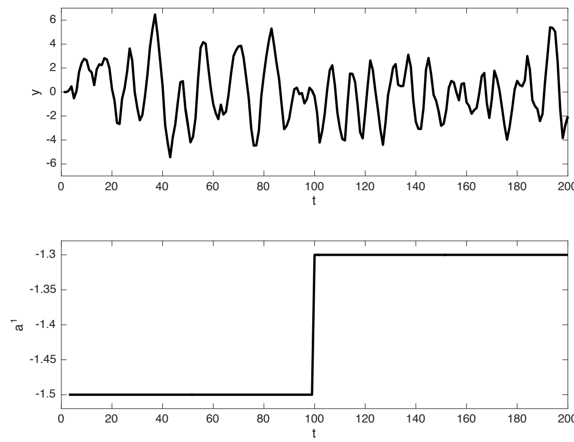

Fig. 3. Data and segmentation results (i.e., the changes in the estimated parameter a^1) for Example 2 when solved using Problem 2 with $\epsilon_2 = 2$.

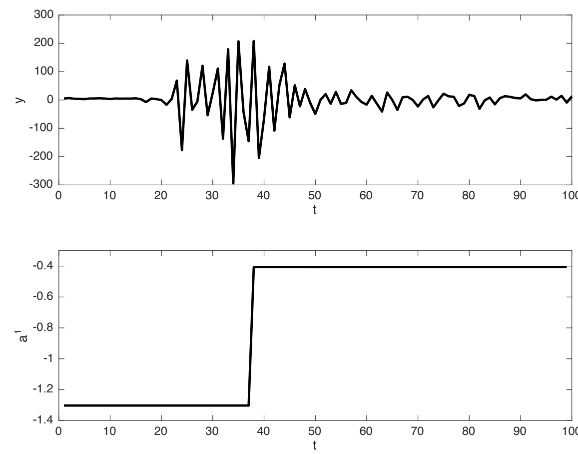

Fig. 4. Data and segmentation results (i.e., the changes in one of the estimated regressor parameter) for Example 3 when solved using Problem 2 with $\epsilon_2 = 2$.

some of which are taken from the System Identification Toolbox [6]. The fitting error function is chosen to be $f(\eta_{t_o}, \eta_{t_1}, \ldots, \eta_N) = \sum_{t=t_o}^{N} \eta_t^2$, corresponding to squared ℓ_2-norm.

Example 1: Consider the model:

$$y_t = -a_t^1 y_{t-1} + b_t^1 u_{t-1} + b_t^2 u_{t-2} + \eta_t.$$

We use the same data provided by [6]. That is, the input $u \in \{-1, 1\}$ is chosen uniformly at random and additive noise η is zero mean with variance 0.1. The parameter $a_t^1 = -0.9$ for all t. For $t < 20$, $b_t^1 = 0$, $b_t^2 = 1$; and for $t \geq 20$, $b_t^1 = 1$, $b_t^2 = 0$. The data and the segmentation results are shown in Figs. 1 and 2 respectively.

Example 2: Consider the time series generated by the following model (without input):

$$y_t = -a_t^1 y_{t-1} - a_t^2 y_{t-2} + \eta_t.$$

Let the additive noise η be zero mean Gaussian with unit variance. At time $t = 100$, the value parameter a_t^1 changes from -1.5 to -1.3, while a_t^2 is identically 0.7 for all times. The data and the segmentation results are shown in Fig. 3.

Example 3: The seismic signal data distributed by MATLAB in `quake.mat` is used in this example. A second order time-varying autoregressive model is used for fitting the data as in [7], with the same preprocessing step. The data and the segmentation results are shown in Fig. 4.

Table I summarizes the results in terms of computation time and compares them to the convex programming approach in [7]. The computations were done on a 3GHz Macbook Pro with 8GB memory. First column shows the computation time when Problem 2 is solved and the second column shows the computation time when $E(m, N)$ is computed for $m = 1, \ldots, N$. Note that by exploring the values of $E(m, N)$ for all m, it is possible to see the whole quality of the fit ($E(m, N)$), model complexity (number of changes m) trade-off space. For comparison with [7], the implementation provided by the authors of [7] is used. In all cases, the proposed dynamic programming algorithm is significantly faster than the method proposed in [7]. Moreover, the solution found by the dynamic programming algorithm is guaranteed to be optimal.

Ex.	DP (known # of switches)	DP (full exploration)	Method of [7]
1	0.08s	0.1s	2.18s
2	0.75s	1.24s	2.98s
3	0.20s	0.40s	10.45s

TABLE I

COMPARISON OF COMPUTATION TIMES OF PROPOSED DYNAMIC PROGRAMMING (DP) BASED APPROACH AND THE APPROACH IN [7].

V. DISCUSSION

Dynamic programming based approach provides an exact and efficient solution to ARX segmentation problem. In particular, it compares favorably to the previously proposed relaxation based approaches [7], [11] when the quality of the fit is measured by ℓ_2-norm. Note that we have not compared the running time to [11] explicitly but the semidefinite programming based relaxation proposed in [11] is known to be computationally much more demanding than the ℓ_1-norm based relaxation approach in [7]. Therefore it can neither be more efficient nor more accurate than the dynamic programming approach proposed here. On the other hand, Piga and Toth [11] present a general method for ℓ_0-norm minimization that has applications beyond ARX segmentation, whereas for the proposed dynamic programming based approach the applications beyond ARX segmentation is not clear.

The ℓ_1-norm based relaxation approach can be easily extended to segmentation of multidimensional (e.g., spatio-temporal) models where the process dynamics depend on more than one indeterminate as shown in [10]. Such models can be used as discrete approximations of PDEs. The ℓ_1-norm based relaxation approach can also be used as an intermediate step in switched system identification when the goal is to find a system with minimum number of modes [9], [10]. Such extensions of the dynamic programming approach presented here are not trivial. It is also worth mentioning that by changing the summation to a maximum operator in the recursion equation (10), dynamic programming approach can handle the case where the quality of the fit is measured by ℓ_∞-norm. However this leads to an algorithm that is computationally less efficient than the greedy exact algorithm proposed for ℓ_∞-norm bounded noise in [9], [10].

VI. CONCLUSIONS

In this paper, we show that even though the problem of segmentation of ARX models is generally non-convex, it admits an efficient and exact solution.

REFERENCES

[1] J. Bai and P. Perron. Computation and analysis of multiple structural change models. *Journal of applied econometrics*, 18(1):1–22, 2003.

[2] R. Bellman and R. Roth. Curve fitting by segmented straight lines. *Journal of the American Statistical Association*, 64(327):1079–1084, 1969.

[3] 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. *Signal Processing Letters, IEEE*, 12(2):105–108, 2005.

[4] R. Killick, P. Fearnhead, and I. Eckley. Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association*, 107(500):1590–1598, 2012.

[5] L. Ljung. System identification: theory for the user. *Prentice Hall information and system sciences series*, 1999.

[6] L. Ljung. The system identification toolbox: The manual, the mathworks. *Inc., MA, USA: Natick*, 2007.

[7] H. Ohlsson, L. Ljung, and S. Boyd. Segmentation of arx-models using sum-of-norms regularization. *Automatica*, 46:1107–1111, 2010.

[8] N. Ozay, M. Sznaier, and O. Camps. Sequential sparsification for change detection. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–6. IEEE, 2008.

[9] N. Ozay, M. Sznaier, C. Lagoa, and O. Camps. A sparsification approach to set membership identification of a class of affine hybrid systems. In *Proc. 47th IEEE Conf. Dec. Control*, pages 123–130, Dec. 2008.

[10] N. Ozay, M. Sznaier, C. Lagoa, and O. Camps. A sparsification approach to set membership identification of switched affine systems. *IEEE Trans. on Automatic Control*, 57(3):634 – 648, march 2012.

[11] D. Piga and R. Toth. An sdp approach for l0-minimization: Application to arx model segmentation. *Automatica*, 49:3646–3653, 2013.