DOI: 10.48011/asba.v2i1.1148

Historical Data Segmentation for System Identification: A Comparison of Methods

Giulio Cesare Mastrocinque Santo*. **Claudio Garcia****

**Telecommunications and Control Systems Department*
Polytechnic School of the University of Sao Paulo, Sao Paulo, SP (Tel: 11982074101; e-mail: giulio.santo@usp.br)
***Telecommunications and Control Systems Department*
Polytechnic School of the University of Sao Paulo, Sao Paulo, SP (Tel: 1130915648; e-mail: clgarcia@lac.usp.br)

Abstract: This article explores the challenging problem of scanning historical data in order to find useful intervals for system identification. A review of the main works related to the subject is presented and two methodologies are described in detail. A single algorithm structure is then presented based on the analyzed methods and practical examples are given, allowing one to understand how to apply the algorithm to massive data.

Resumo: Este artigo aborda o desafiador problema de realizar uma varredura em dados históricos com o objetivo de encontrar intervalos adequados para a realização de uma identificação de sistema. Uma revisão dos principais trabalhos sobre o assunto é inicialmente apresentada e, então, duas metodologias são descritas detalhadamente. Em seguida, uma estrutura de algoritmo é apresentada com base nos métodos analisados e exemplos práticos são fornecidos, permitindo o entendimento de como o algoritmo pode ser aplicado em dados históricos massivos.

Keywords: Condition Number; Data Mining; Dynamic Response; Historical Data; Numerical Analysis; System Identification.

Palavras-chaves: Número de Condição; Mineração de Dados; Resposta Dinâmica; Dados Históricos; Análise Numérica; Identificação de Sistemas.

1. INTRODUCTION

System Identification is a set of techniques used to mathematically model the dynamics of a system. The ability to create industrial process models is extremely relevant for industry, once it allows the development of different activities that aggregate value to companies, such as designing advanced controllers (like Model Predictive Controllers), providing optimal tuning of Proportional Integral Derivative (PID) controllers, developing training simulators, detecting possible system failures, checking for process quality and performing predictive maintenance.

Although System Identification is entirely dependent on data, it does not take advantage of the data management infrastructure available in many companies. This is because, most of the time, the data used to model an industrial process is collected by physical experiments that excite the process variables and create a dynamic response. The execution of these experiments is usually undesirable, since there is a cost to deviate the process from its production operating condition and the fact that experiments can take a long time to be successfully completed. It is quite natural, therefore, the idea to use the large amount of historical process data available in many companies to accomplish this task.

Using historical measured data to automatically find useful models of an industrial process is not a trivial task and few works in the literature directly address this problem.

The problem was explicitly introduced by the first time in (Peretzki *et al*., 2011), where an approach based on the Laguerre Model structure was used to find suitable intervals for system identification in closed-loop systems. A more detailed version of this work was published in (Bittencourt *et al*., 2015). A similar approach can be found in (Shardt; Huang, 2013a), but using an Autoregressive Moving Average with Exogenous Inputs (ARX) model structure.

In (Shardt; Huang, 2013b), statistical properties of the discrete-time signal entropy were studied and a change detection index was proposed to perform the segmentation of time series data. This work was used in (Shardt; Shah, 2014) as an additional step to the method proposed in (Peretzki *et al*., 2011), where a differential entropy between the input and the output is used to find similar segments of excitation. In (Ribeiro; Aguirre, 2015), a method based on the Autoregressive (AR) structure was proposed using routine operating data.

The multivariable problem was introduced in (Patel, 2016), where an extension of the works in (Peretzki *et al*., 2011) and in (Bittencourt *et al*., 2015) is proposed to include Multiple

Input Multiple Output (MIMO) systems in the analysis of the open-loop identification scenario. The multivariable problem for closed-loop systems was addressed in (Arengas; Kroll, 2017a). Moreover, in (Arengas; Kroll, 2017b), an open-loop analysis was also presented using the ARX structure.

In (Wang *et al*., 2018), a new method to search data segments suitable for system identification was presented, totally based on statistical methods, in which the authors applied a top-down approach to detect change-points in the data.

Finally, in (Shardt; Brooks, 2018), the problem of searching intervals for MIMO systems in closed-loop mode and using operating data was addressed.

Herein, two methods are compared based on the reviewed literature. More specifically, the works published in (Peretzki *et al*., 2011), (Shardt; Huang, 2013a), (Bittencourt *et al*., 2015), (Ribeiro; Aguirre, 2015) and (Arengas; Kroll, 2017a) are summarized in two methods, both using numerical conditioning as the basis to validate adequate time intervals to obtain a process model operating in closed-loop control. Moreover, the top-down change-point detection presented in (Wang *et al*., 2018) is proposed as an optional first step of the explained methods. Finally, a generic algorithm structure is presented and its use is exemplified through data from the step response of a pH neutralization process.

This paper is organized as follows: the mathematical background necessary to understand the methods that are compared is described in Section 2; then, two different segmentation methods are presented in Section 3 based on the reviewed literature, where they are combined into a single algorithm; finally, examples of application are given in Section 4, with conclusions being drawn on Section 5.

2. MATHEMATICAL BACKGROUND

2.1 Open-loop and Closed-loop Identification

Consider a closed-loop system as in Fig. 1. Notice that $C(q)$ is the controller transfer function; $G(q)$ is the process model; $H(q)$ is the disturbance model; $r(k)$ is the process set point; $u(k)$ is the output of the controller (manipulated variable); $d(k)$ is a measured disturbance; $v(k)$ is measurement white noise and $y(k)$ is the process output (controlled variable).

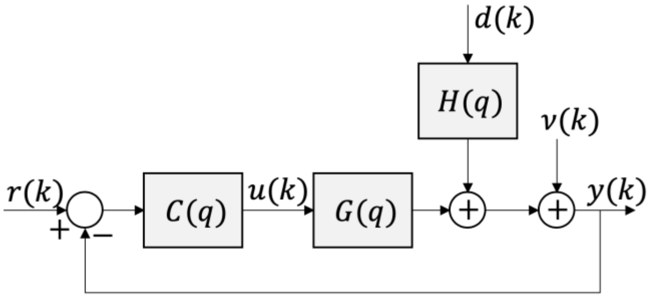

Fig. 1. Closed-loop system (Adapted from Wang *et al*., 2018).

A closed-loop identification is commonly known as the process to obtain the model $G(q)$ when the system contains a feedback control loop, as in Fig. 1. Different identification approaches can be used in this scenario, some of them are described in (Ljung, 1999), such as the “Direct Approach” and the “Indirect Approach”. However, one could also obtain a model of the system considering the set point and the output variable, in which case the resulting model includes the controller transfer function $C(q)$. In this work, a model of the process $G(q)$ is the ultimate goal.

Notice that to identify a model with the process operating in closed-loop, if the control algorithm is complex enough, a process model can be obtained in the presence of disturbances, as detailed in (Bittencourt *et al*., 2015).

The necessary conditions to obtain a process model using an ARX structure for systems operating in closed-loop control are studied in (Shardt; Huang, 2013a). This problem is also addressed in (Bittencourt *et al*., 2015) and it is shown, in a general manner, that if the set point signal is persistently exciting, one has enough information to obtain a model through closed-loop identification. For this reason, in this paper, potential intervals with meaningful excitation in both the set point and in the output variables are first found. Then, the manipulated variable and the output variable are used to evaluate the numerical properties of these intervals.

2.2 Least Squares Problem

Industrial processes are usually modeled through time series regression models. A common approach to estimate the parameter vector of a particular regression model is through the prediction error methods. Let us assume a generic Linear Regression model structure as follows (Aguirre, 2015):

$$y(k) = \psi_{yu}^{T}(k-1)\hat{\theta} + \xi(k). \tag{1}$$

The prediction error in this case is defined as:

$$\xi\big(k, \hat{\theta}\big) = y(k) - \psi_{yu}^{T}(k-1)\hat{\theta}. \tag{2}$$

where $\psi_{yu}^{T}(k-1)\hat{\theta}$ is the regression prediction using information available until the discrete-time instant $k-1$. One can also represent this prediction as $\hat{y}(k|k-1) = \psi_{yu}^{T}(k-1)\hat{\theta}$, which is called the one step ahead prediction (Aguirre, 2015).

Because the objective of this work is to find intervals of data suitable for system identification, let us consider a data sample of length N_s. So, for this sample, an estimation $\hat{\theta}_{N_s}$ of the regression parameter vector can be obtained through the solution of the Least Squares Problem, defined as (Aguirre, 2015):

$$\hat{\theta}_{N_s} = \arg\min_{\hat{\theta}} \sum_{k=1}^{N_s} \big[y(k) - \psi_{yu}^{T}(k-1)\hat{\theta}\,\big]^2. \tag{3}$$

A closed solution for Equation (3), considering this data sample, can be obtained as (Aguirre, 2015):

$$\hat{\theta}_{N_s} = \left[\frac{1}{N_s}\sum_{k=1}^{N_s} \psi_{yu}(k-1)\,\psi_{yu}^{T}(k-1)\right]^{-1} \times \tag{4}$$

$$\left[\tfrac{1}{N_s}\textstyle\sum_{k=1}^{N_s} \psi_{yu}(k-1)\,y(k)\right] .$$

It is important to say that the solution to the Least Squares Problem is only feasible if the inverse of matrix $\hat{R}_{N_s} = \frac{1}{N_s}\sum_{k=1}^{N_s} \psi_{yu}(k-1)\,\psi_{yu}^T(k-1)$ exists. This is a symmetric and positive definite matrix, frequently called the **Information Matrix.**

2.3 Laguerre and Autoregressive (AR) Model Structures

2.3.1 Autoregressive (AR) Structure

The autoregressive model is a linear structure composed of delayed observations of the output variable. The difference equation for this structure is $y(k) - a_1 y(k-1) - \cdots - a_{n_y} y(k-n_y) = \nu(k)$, where n_y is the model order (Aguirre, 2015). The regressor vector and the parameter vector for this model are:

$$\psi_{yu}^T(k-1) = \left[y(k-1) \ldots y(k-n_y)\right]. \tag{5}$$

$$\hat{\theta} = \left[\hat{a}_1 \ldots \hat{a}_{n_y}\right]^T. \tag{6}$$

2.3.2 Laguerre Structure

The Laguerre structure was first proposed for system identification in (Wahlberg, 1991). While the AR structure only considers delayed versions of the output, the Laguerre structure only considers filtered versions of the input variable $u(k)$, as follows:

$$y(k) = \sum_{i=1}^{n_b} \bar{g}_i L_i(q,\alpha) u(k). \tag{7}$$

where $L_i(q,\alpha) = \frac{\sqrt{(1-\alpha^2)}}{q-\alpha}\left(\frac{1-\alpha q}{q-\alpha}\right)^{i-1}$ is the Laguerre Filter, α is the Laguerre filter pole and $\bar{g}_i$ are the regressor parameters.

It is interesting to mention that the Laguerre Structure is implicitly capable of estimating the time delay of the system. As explained in (Peretzki *et al.*, 2011), the maximum delay $\bar{d}$ that can be incorporated in this model structure is $\bar{d} = \frac{-2(n_b-1)T_s}{\log(\alpha)}$, with n_b being the model order and T_s the sampling period.

Based on the reviewed literature and on the experiments performed by the authors when mining historical data, typical values for the Laguerre Filter order and pole are chosen in the following ranges: $n_b \in [8, 10]$ and $\alpha \in [0.8, 0.95]$. Moreover, Laguerre poles too close to 1 produce information matrices with very large condition numbers, as well shown in (Patel, 2016).

The regressor vector and the parameters vector for this structure are defined bellow:

$$\psi_{yu}^T(k) = \left[L_1(q,\alpha)u(k) \ldots L_{n_b}(q,\alpha)u(k)\ \right]. \tag{8}$$

$$\hat{\theta} = \left[\bar{g}_1 \ldots \bar{g}_{n_b}\right]^T. \tag{9}$$

2.3.3 Regressor Matrix

Given a regression structure such as the AR or the Laguerre, a regressor matrix for an interval of data of length N_s can be defined as (Aguirre, 2015):

$$\psi_{\mathrm{N_s}} = \begin{bmatrix} \psi_1(k) & \psi_2(k) & \cdots & \psi_{n_\theta}(k) \\ \vdots & \vdots & \ddots & \vdots \\ \psi_1(k+N_s) & \psi_2(k+N_s) & \cdots & \psi_{n_\theta}(k+N_s) \end{bmatrix}. \tag{10}$$

where $\psi_{\mathrm{N_s}} \in \mathbb{R}^{N_s \times n_\theta}$. In this case, Equation (4) can be rewritten as:

$$\hat{\theta}_{N_s} = \left[\psi_{\mathrm{N_s}}^{\mathrm{T}}\psi_{\mathrm{N_s}}\right]^{-1}\psi_{\mathrm{N_s}}^{\mathrm{T}}y. \tag{11}$$

2.4 QR-Decomposition

A numerically more stable solution to the Least Squares Problem is obtained in (Peretzki *et al.*, 2011) through the so-called QR-decomposition, which consists of decomposing a matrix $A \in \mathbb{R}^{m \times n}$ in the form $A = QR$, with matrix $Q \in \mathbb{R}^{m \times m}$ being orthogonal and matrix $R \in \mathbb{R}^{m \times n}$ being upper triangular (Verhaegen; Verdult, 2007).

As explained in (Peretzki, *et al.* 2011), the Least Squares Problem can be solved through the QR-decomposition of matrix $A = \left[\psi_{\mathrm{N_s}}\ Y\right]$, with $Y \in \mathbb{R}^{N_s}$ being the output sample and $\psi_{\mathrm{N_s}} \in \mathbb{R}^{N_s \times n_\theta}$ being the regressor matrix for a data sample of length N_s and for a given model structure of order n_θ. Matrix R can then be written as follows (Peretzki, *et al.* 2011):

$$R = \begin{bmatrix} R_0 \\ \vdots \\ 0 \end{bmatrix}, \quad R_0 = \begin{bmatrix} R_1 & R_2 \\ 0 & R_3 \end{bmatrix}. \tag{12}$$

with $R_0 \in \mathbb{R}^{N_s \times (n_\theta+1)}$, $R_1 \in \mathbb{R}^{n_\theta \times n_\theta}$, $R_2 \in \mathbb{R}^{n_\theta \times 1}$ and R_3 being a scalar value.

As shown in (Peretzki, *et al.* 2011), the Least Squares Problem can be solved using the QR-decomposition through $R_1\hat{\theta} = R_2$, being the Information Matrix rewritten as:

$$\hat{R}_{N_s} = \frac{1}{N_s}\psi^T\psi = \frac{1}{N_s}R_1^T R_1. \tag{13}$$

2.5 Singular Value Decomposition

The Singular Value Decomposition consists of transforming a matrix $A \in \mathbb{R}^{m \times n}$ into the form $A = U\Sigma V^T$, with $U \in \mathbb{R}^{m \times m}$ and $V \in \mathbb{R}^{n \times n}$ being orthogonal matrices. The elements of matrix $\Sigma \in \mathbb{R}^{m \times n}$ are all zero, except for the diagonal elements σ_i, which are ordered as $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_k = 0$. (Verhaegen; Verdult, 2007). Notice that $r = \mathrm{rank}(A)$ and that $k = min(m,n)$. The singular values of matrix A can be defined as the diagonal elements σ_i of matrix Σ. As will be clear in this paper, singular values can be very handful for numerically computing the rank of a matrix.

3. SEGMENTATION METHODS

3.1 Method 1: A Condition Number Approach

The method presented in this subsection is based on (Peretzki *et al.*, 2011), (Shardt; Huang, 2013a), (Bittencourt *et al.*, 2015) and (Arengas; Kroll, 2017a). In this paper, the method is broken down into three main steps, following a structure similar to the one proposed in (Patel, 2016).

3.1.1 Step 1: Finding potential intervals through the search of meaningful excitations

When one is looking at historical process data, most of the data are not useful for system identification. In fact, the majority of the data are in steady state or disturbed by noise. This is why the first step of the method is to find moments where the signal was significantly "active" and so, could be useful to find a dynamic model of the system. In order to find these initial intervals of excitation, an Exponentially Weighted Moving Average (EWMA) filter is applied in (Peretzki, *et al*. 2011), as shown below:

$$\mu_x(k) = \lambda_\mu \times x(k) + \left(1 - \lambda_\mu\right) \times \mu_x(k-1). \tag{14}$$

$$S_x(k) = \frac{2-\lambda_\mu}{2} - (\lambda_S \times (x(k) - \mu_x(k))^2 + (1 - \lambda_S) \times S_x(k-1)). \tag{15}$$

where $\mu_x(k)$ is an estimate of the signal average and $S_x(k)$ is an estimate of the signal variance. Moreover, λ_μ and λ_S are, respectively, the mean and the variance exponential forgetting factors.

Potential intervals can then be found setting a threshold to the estimated variance, such that every point that satisfies $S_x(k) > l_s$ is marked as useful. In this work, the set point and the output variables are the ones investigated. In fact, the existence of excitation in the set point is considered a requirement, once the closed-loop identification case is being considered.

3.1.2 Step 2: Assessing the Numerical Conditioning of the Potential Intervals

The second natural step of the algorithm consists of evaluating if the potential intervals obtained by Step 1 are valid candidates for system identification. A given interval of length N_s is considered useful for system identification if the Information Matrix $\hat{R}_{N_s}$, defined in Equation (13), for this interval is numerically well conditioned and, thus, could be inverted to solve the least squares problem in Equation (11) (Bittencourt *et al.*, 2015).

In this method, the regressor matrix is obtained by the Laguerre Filter structure as defined in Equations (7) - (9). To assess the numerical conditioning of the information matrix, it is proposed in (Peretzki, *et al*. 2011) the computation of its Condition Number.

The $2-norm$ condition number is considered in this method, being defined as follows:

$$\kappa_2\left(\hat{R}_{N_s}\right) = \frac{\sigma_{max}(\hat{R}_{N_s})}{\sigma_{min}(\hat{R}_{N_s})}. \tag{16}$$

where $\sigma_{max}(\hat{R}_{N_s})$ and $\sigma_{min}(\hat{R}_{N_s})$ are, respectively, the maximum and the minimum singular values of $\hat{R}_{N_s}$, as defined in Subsection 2.5. The closer is the $2-norm$ condition number to 1, the better is the numerical conditioning of $\hat{R}_{N_s}$. On the contrary, the greater is the condition number, more poorly conditioned is the matrix and less reliable is the solution of the least squares problem (Bittencourt *et al.,* 2015).

Therefore, $\kappa_2\left(\hat{R}_{N_s}\right)$ must be compared to a given threshold, such that if a particular segment satisfies the criterion $\kappa_2\left(\hat{R}_{N_s}\right) < l_\kappa$, the interval is still a potential interval and the algorithm can move to Step 3.

3.1.3 Step 3: Checking if the Input and the Output are Correlated

The final step of this method is to verify, for every interval that met the criteria in Steps 1 and 2, if the input and the output signals are correlated. The way this test is proposed in the reviewed works is through the computation of an estimate $\hat{\theta}_{N_s}$ of the parameter vector, followed by a statistical validation if it is sufficiently different from zero. This is done through a chi-squared statistical test.

An estimate $\hat{\theta}_{N_s}$ of the parameters vector can be obtained using the QR-decomposition as follows (Peretzki, *et al*. 2011):

$$\hat{\theta}_{N_s} = R_1^{-1} R_2. \tag{17}$$

The chi-squared critical value is then computed as in (LJUNG, 1999):

$$\hat{\chi}_{N_s} = \hat{\theta}_{N_s}^T \hat{P}_{N_s}^{-1} \hat{\theta}_{N_s} \in \chi_d. \tag{18}$$

where $\hat{P}_{N_s}$ is the covariance matrix and $d = n_\theta$ is the degree of freedom of the statistical test. Using the QR-decomposition, Equation (18) can be rewritten as (Peretzki, *et al*. 2011):

$$\hat{\chi}_{N_s} = \left\| \frac{R_2\sqrt{N_s}}{|R_3|} \right\|_2^2. \tag{19}$$

Therefore, for a given statistical significance level α, if the computed statistic $\hat{\chi}_{N_s}$ is greater than its critical value $\chi_{d,\alpha}$ ($\hat{\chi}_{N_s} > \chi_{d,\alpha}$), the interval is finally considered suitable for system identification.

3.2 Method 2: An Effective Ranking Method

An alternative approach can be found in (Ribeiro; Aguirre, 2015) and it is used as the basis of this method, where the AR structure is used instead. Although, in this paper, this method is also divided into three steps, Step 1 is considered the same as the one in the previous subsection and only Steps 2 and 3 are elucidated.

3.2.1 Step 2: Assessing the Numerical Conditioning of the Potential Intervals

As an alternative to compute the condition number of the information matrix, this method computes the effective rank. One can understand the effective rank as a numerical estimation for the actual rank of a matrix. Therefore, for a matrix $A \in \mathbb{R}^{m \times n}$, the maximum value of the effective rank is $min(m, n)$, in which case the matrix is considered full effective rank (Ribeiro; Aguirre, 2015).

Two different ways to compute the effective rank are proposed in (Ribeiro; Aguirre, 2015). The first way is through the normalized singular values p_i:

$$p_i = \frac{\sigma_i}{\|\sigma\|_1} \qquad \|\sigma\|_1 = \sum_{i=1}^{k} |\sigma_i|. \tag{20}$$

$$r_1^{ef} = \sum_{i=1}^{k} H[p_i - l_1]. \tag{21}$$

where $H(\cdot)$ is the Heaviside Function and l_1 is the singular value tolerance. Here, p_i is computed as in (Roy; Vetterli, 2007).

The second proposed way of computing the effective rank is by the difference of two consecutives singular values, as shown below:

$$r_2^{ef} = \sum_{i=2}^{k} H[(\sigma_{i-1} - \sigma_i) - l_2]. \tag{22}$$

Notice that the effective rank r^{ef} is an integer value that corresponds to the numerical estimate of the actual rank of a matrix.

As in the previous method, for each potential interval, the effective rank of the Information Matrix can be computed and associated with a threshold, such that if $r^{ef}\left(\hat{R}_{N_s}\right) > l_{efr}$ for a particular interval, one can move to Step 3.

3.2.2 Step 3: Checking if the Input and the Output are Correlated

Because the cross-correlation of two signals is a function of the lag τ between them, in (Ribeiro; Aguirre, 2015) it is proposed a singular scalar metric for verifying if the input and the output are actually correlated, which is computed as follows:

$$s = \sum_{\tau = -\tau_{max}}^{\tau = \tau_{max}} g(\rho(\tau), \tau, p). \tag{23}$$

$$g(\rho(\tau), \tau, p) = \begin{cases} 0, & \text{if } |\rho| \leq p \\ \dfrac{|\rho(\tau) - p|}{|\tau|}, & \text{if } |\rho| > p \text{ and } \tau \neq 0 \\ |\rho(\tau)| - p, & \text{if } |\rho| > p \text{ and } \tau = 0 \end{cases}. \tag{24}$$

where $\rho(\tau)$ is the normalized cross-correlation function for a particular lag τ, τ_{max} is the maximum lag of interest and $[-p, +p]$ defines the 95% confidence interval, with $p = 1.96/\sqrt{N_s}$ (Ribeiro; Aguirre, 2015).

Notice that s is a measurement of how greater the normalized cross-correlation is from the limits of a 95% confidence interval in the lag range $[-\tau_{max},\ \tau_{max}]$. Naturally, the greater is s, the greater is the correlation between the two signals. Again, s can be associated with a threshold l_s such that if $s > l_{cc}$, the signals can be assumed to be enough correlated. As in the previous method, if a particular interval meets the criteria in the three steps, it is considered adequate to obtain a dynamic model of the system under study.

3.3 A Change-point Algorithm

Another way to find the potential intervals in Step 1 of the previous methods is proposed in (Wang *et al.*, 2018 apud Pettitt, 1979) through a top-down, non-parametric change-point detection algorithm. A top-down algorithm is one that starts with the entire dataset and divides it until no further divisions can be found.

As explained in (Wang *et al.*, 2018), let us initially consider a data segment of length N_s as $x(k{:}k + N_s - 1) = (x(k), x(k+1), \dots, x(k + N_s - 1))$. Therefore, in this method, the initial segment begins with the entire data: $(x(0), x(1), \dots, x(N))$, being N its length. Notice that the segment of length N_s is a subset of the entire dataset of length N.

Then, the relative position of all points in a particular segment of length N_s can be calculated as (Wang *et al.*, 2018):

$$D(t) = \sum_{j=k}^{k+N_s-1} \text{sign}(x(t) - x(j)) \text{ for } t = k, \dots, k + N_s - 1. \tag{25}$$

with $\text{sign}(\cdot)$ being the signal function. The cumulative sum of $D(t)$ is then calculated as $C(t) = C(t-1) + D(t)$, for $t = k, \dots, k + N_s - 1$, considering the initial value $C(k-1) = 0$ (Wang *et al.*, 2018).

Finally, a change-point τ^* is defined as a time index that maximizes the absolute value of $C(t)$. A hypothesis test is formulated in (Wang *et al.*, 2018) assuming the following null hypothesis:

$$H_0{:} \arg \max_{k \leq \tau^* \leq k + N_s - 1} |C(t)| \text{ is not a change point.}$$

The $p -$ value associated with this hypothesis test is defined as (Wang *et al.*, 2018 apud Pettitt, 1979):

$$p = 2e^{\left(\frac{-6|C(\tau^*)|^2}{N_s^2 + N_s^3}\right)}. \tag{26}$$

Therefore, for a given significance level α, τ^* is considered a change-point index if $p < \alpha$. If a change-point is found, the current data segment must be split into two new segments divided by this change-point (Wang *et al.*, 2018):

$$\begin{cases} x(k{:}\tau^*) = (x(k), \dots, x(\tau^*)) \\ x(\tau^* + 1{:}N) = (x(\tau^* + 1), \dots, x(N)) \end{cases}$$

This process is then iterated and the data is further divided until no more change-points can be found within the significance level α.

3.4 Outline of the Algorithms

Based on the methods explained in the previous subsections and in the referenced works, an algorithm to automatically detect intervals suitable for system identification for the closed-loop scenario is formulated as in Fig. 2, which contains elements of the algorithms proposed in (Peretzki, *et al.* 2011), (Ribeiro; Aguirre, 2015), (Patel, 2016) and (Wang *et al.*, 2018).

An alternative implementation of this method would be to allow the initial intervals to be incremented in time. In a similar way to what is done in (Arengas; Kroll, 2017a), instead of only checking if the intervals from Step 1 meet the Steps 2 and 3 criteria, these initial intervals could be incremented in time until the criteria in Steps 2 and 3 are still being met.

Notice that this algorithm could be easily adapted to the open-loop identification scenario. The modification would be to look for "active" manipulated variables instead of looking for "active" set points in Step 1.

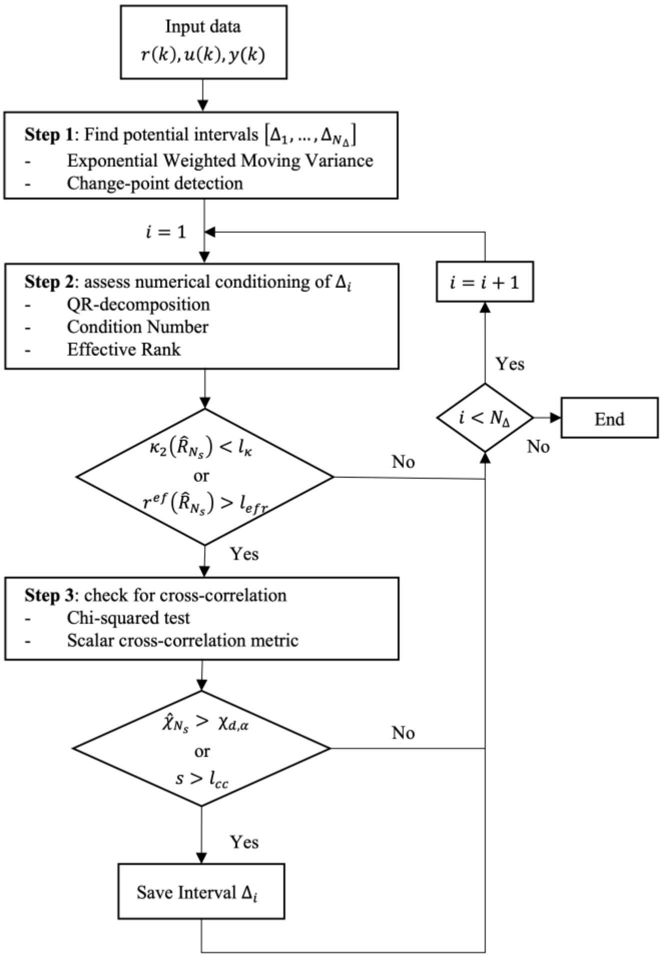
Input data
$r(k), u(k), y(k)$
Step 1: Find potential intervals $[\Delta_1, \dots, \Delta_{N_\Delta}]$
- Exponential Weighted Moving Variance
- Change-point detection
$i = 1$
Step 2: assess numerical conditioning of Δ_i
- QR-decomposition
- Condition Number
- Effective Rank
$i = i + 1$
Yes
$i < N_\Delta$
End
No
$\kappa_2(\hat{R}_{N_s}) < l_\kappa$ or $r^{ef}(\hat{R}_{N_s}) > l_{efr}$
No
Yes
Step 3: check for cross-correlation
- Chi-squared test
- Scalar cross-correlation metric
$\hat{\chi}_{N_s} > \chi_{d,\alpha}$ or $s > l_{cc}$
No
Yes
Save Interval Δ_i

Fig. 2. Algorithm structure to find intervals suitable for system identification.

4. EXAMPLES

In order to exemplify how the presented algorithm works and how it can be applied to massive data, a pH Neutralization Plant from the Industrial Process Control Laboratory from University of São Paulo is used. A simplified P&ID of the system can be seen in Fig. 3. Basically, the process consists of two circuits: one for the acid solution and another for the base solution. The two solutions are mixed in a reaction tank, where the chemical reaction occurs. In this plant, only the flow rate of the base solution is manipulated to control the pH value.

The temperature, the pH and the level variables are strongly decoupled, such that only the pH variable will be used as a SISO closed-loop system.

The data that are used consists of two consecutive step changes in the pH set point around the operating value of 2.8 pH. Moreover, before applying the algorithms, the data are normalized around the range $[-0.5, +0.5]$ and a low-pass filter is applied to reduce high frequency noise.

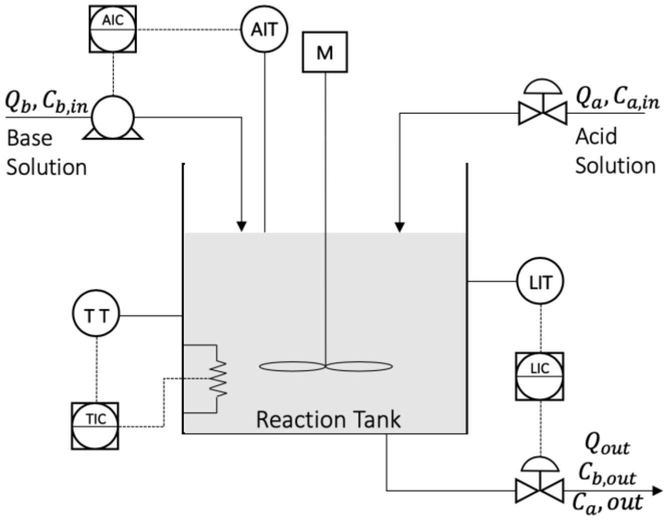

Fig. 3. Simplified P&ID of the pH Neutralization Process.

4.1 *Finding Initial Intervals*

As shown in Fig. 2, the first step of the algorithm is to find initial intervals to be evaluated as adequate to perform system identification. If one uses the change-point method described in Subsection 3.3, two values must be chosen: the statistical significance value and the minimum length accepted for each interval. Assuming a significance level $\alpha = 0.01$ and a minimum interval length of 1500 data points, the data would be divided according to Fig. 4, where the dots are the change-point positions, the blue curve is the output variable and the orange curve is the set-point variable.

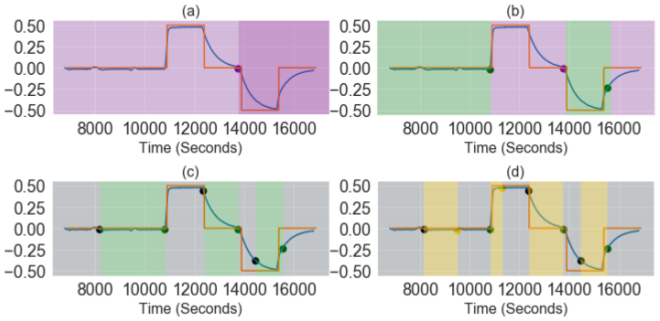

Fig. 4. Change-point detection algorithm. (a) Iteration 1, (b) Iteration 2, (c) Iteration 3, (d) Iteration 4.

Because this is a top-down algorithm, one can notice that the entire dataset is divided, where the final intervals are those in Fig. 4 (d). This implies that all intervals must be evaluated through Steps 2 and 3 of the algorithm. Moreover, the computational complexity of this algorithm is about $O(N^2)$ for a dataset of length N (Wang *et al.*, 2018), which is very heavy for massive data.

If one now applies the EWMA filter described in Subsection 3.1 for different values of the forgetting factors λ_μ and λ_S, the filtered set point and output signals in Fig. 5 could be obtained.

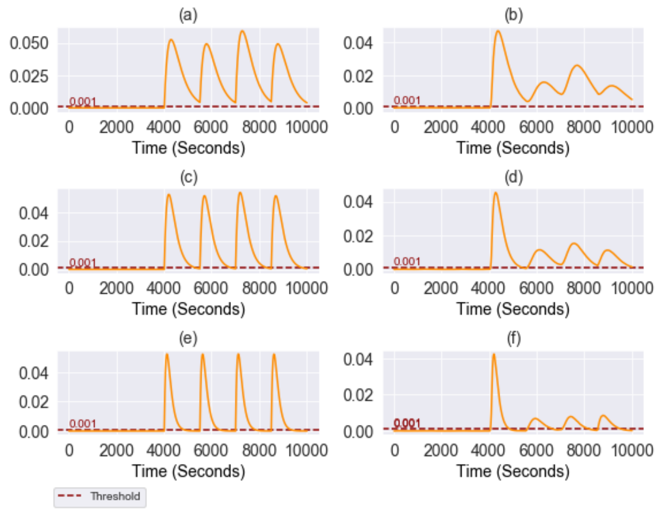

Fig. 5. (a) pH set point variance for $\boldsymbol{\lambda_S, \lambda_\mu = 0.002}$, (b) output pH variance for $\boldsymbol{\lambda_S, \lambda_\mu = 0.002}$, (c) pH set point variance for $\boldsymbol{\lambda_S, \lambda_\mu = 0.003}$, (d) output pH variance for $\boldsymbol{\lambda_S, \lambda_\mu = 0.003}$, (e) pH set point variance for $\boldsymbol{\lambda_S, \lambda_\mu = 0.005}$, (f) output pH variance for $\boldsymbol{\lambda_S, \lambda_\mu = 0.005}$.

It is interesting to notice how sensitive the algorithm is to the choice of the forgetting factors, as well as to the choice of the associated thresholds. If the threshold of 0.001 is chosen for the forgetting factors $\boldsymbol{\lambda_S, \lambda_\mu = 0.002}$, one can notice that the entire step response is considered. On the other hand, using the same threshold for forgetting factors $\boldsymbol{\lambda_S, \lambda_\mu = 0.005}$ results in the initial intervals of Fig. 6.

Notice that λ_S and λ_μ were chosen with the same values to simplify their choices. However, individual values could be provided to each forgetting factor, although the authors have not found advantages in doing so in a practical context.

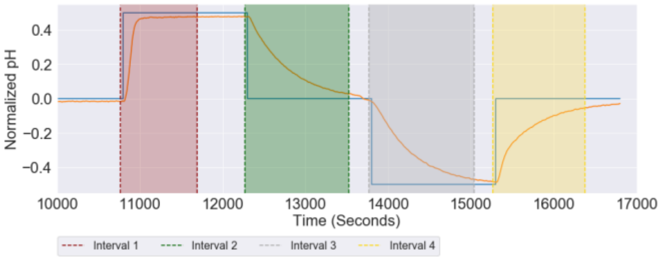

Fig. 6. Initial intervals obtained with $\boldsymbol{\lambda_S, \lambda_\mu = 0.005}$.

Finally, it is interesting to point out that the EWMA filter only selects "active" signals, while the change-point method returns the entire partitioned dataset.

4.2 *Assessing the Numerical Conditioning of the Initial Intervals*

To understand how each interval would be evaluated against its numerical conditioning, let us consider the four intervals from Fig. 6. Let us now apply a Laguerre Filter with pole $\alpha = 0.95$ and order $N_b = 10$ to each interval to compute the condition number and the chi-squared test. In the same fashion, let us consider an AR structure of order $n_y = 100$ to compute the effective rank and the scalar cross-correlation metric. For the effective rank, a tolerance of $l_1 = 1E - 9$ is used; for the scalar cross-correlation metric, the delay range considered is $[-20, 20]$.

Table 1 summarizes the resulting Condition Number and Effective Rank values for each potential interval.

Table 1. Condition number and effective rank values for each potential interval.

		Intervals			
Method	Structure	**1**	**2**	**3**	**4**
Condition Number	Laguerre	$7.2E4$	$2.6E4$	$3.1E4$	$1.0E5$
Effective Rank	AR	37	50	42	61

In the same way, Table 2 summarizes the resulting Chi-squared and scalar cross-correlation values.

Table 2. Chi-squared and scalar cross-correlation values for each potential interval.

		Intervals			
Method	Structure	**1**	**2**	**3**	**4**
Chi-squared	Laguerre	80.4	158.9	176.9	33.9
Cross-Correlation	AR	7.96	7.75	6.82	4.78

The chi-squared critical value for a significance level of $\alpha = 0.00001$ and degree of freedom $d = N_b = 10$ is 41.3, which is higher than the value obtained for Interval 4. Moreover, clearly Intervals 2 and 3 have the highest chi-squared value, which suggests the highest cross-correlation value.

If one considers the thresholds for the condition number, effective rank and cross-correlation values, respectively, as $l_\kappa = 35000$, $l_{efr} = 45$, $l_s = 6.5$, only Intervals 2 and 3 would be considered through both the condition number and the effective rank methods.

In order to verify if the results above are consistent, an ARX structure of input order $N_u = 6$, output order $N_y = 14$ and dead time order $N_k = 4$ was used to model all intervals, considering Interval 2 and Interval 3 alternately as the validation data. To compare the results, MATLAB® FIT metric is used, which can be defined as:

$$\text{FIT} = 100\left(1 - \frac{\|y-\hat{y}\|}{\|y-\bar{y}\|}\right). \qquad (27)$$

Considering a 100 steps-ahead prediction, results in Table 3 are obtained. Notice that training intervals are never validated with themselves. As expected, Interval 1 had the worst performance and could not result in a model at all. Intervals 2 and 3, which had the lowest condition number and the highest chi-squared values, had a positive fit value, with Interval 3 having the best performance of all. It is also interesting to point out that, although Interval 4 had the highest condition number, it also had the highest effective rank computed with the AR structure. If one looks at the FIT value of Interval 4, it was also positive. However, for both validation scenarios, this interval had lower performances compared to Intervals 2 and 3, which can probably be explained by its low cross-correlation and chi-squared values.

Table 3. Comparison of the cross-validation FIT values for each potential interval.

	Training Intervals			
Validation Intervals	**1**	**2**	**3**	**4**
2	-126.1	----------	61.57	56.3
3	-489.7	35.14	-----------	27.78

5. CONCLUSIONS

Different works concerning the problem of searching intervals suitable for system identification in historical data are introduced. A few of these works are then explained in detail and used to present two methodologies based on numerical conditioning.

From the examples presented, one can notice that the proposed algorithm is very susceptible to the parameter's choice. In a first moment, different choices of forgetting factors λ_μ and λ_S and its associated thresholds, or different values of α for the change-point detection algorithm, can lead to completely different results, both in the number of intervals and in their sizes.

The choice of the condition number or the effective rank thresholds is also very delicate, since these values may not vary much for similar intervals. Moreover, the threshold for determining acceptable cross-correlation values must also be provided, which result in a large set of parameters to be chosen. In fact, the choice of parameters becomes especially hard in massive data, where visualization of the data is not possible.

However, the presented algorithm is clearly able to find sufficiently "active" data and evaluate whether this data can be useful to estimate models. In addition, if multiple intervals are obtained, one can take advantage of this fact to choose the ones that result in better models.

REFERENCES

Aguirre, L. A. (2015). *Introdução à Identificação de Sistemas: técnicas lineares e não lineares: teoria e aplicação*. 4. ed. Belo Horizonte, Brasil: Editora UFMG.

Arengas, D., Kroll, A. (2017a). A Search Method for Selecting Informative Data in Predominantly Stationary Historical Records for Multivariable System Identification. In *Proceedings of the 21st International Conference on System Theory, Control and Computing (ICSTCC)*. Sinaia, Romenia: IEEE, p. 100–105.

Arengas, D., Kroll, A. (2017b). Searching for informative intervals in predominantly stationary data records to support system identification. In *Proceedings of the XXVI International Conference on Information, Communication and Automation Technologies (ICAT)*. Sarajevo, Bosnia-Herzegovina: IEEE.

Bittencourt, A. C. *et al*. (2015). An algorithm for finding process identification intervals from normal operating data. Processes, v. 3, p. 357–383.

Ljung, L. (1999). *System Identification: Theory for User*. 2. ed. Upper Saddle River, New Jersey: Prentice-Hall, Inc.

Patel, A. (2016). *Data Mining of Process Data in Mutlivariable Systems*. Degree project in electrical engineering — Royal Institute of Technology, Stockholm, Sweden.

Peretzki, D. *et al*. (2011). Data mining of historic data for process identification. In *Proceedings of the 2011 AIChE Annual Meeting*, p. 1027–1033.

Pettitt, A. N. A. (1979). A non-parametric approach to the change-point problem. *Journal of the Royal Statistical Society*, v. 28, n. 2, p. 126–135.

Ribeiro, A. H., Aguirre, L. A. (2015). Selecting transients automatically for the identification of models for an oil well. *IFAC-PapersOnLine*, v. 48, n. 6, p. 154–158.

Roy, O., Vetterli, M. (2007). The effective rank: A measure of effective dimensionality. In *Proceeding of the 15th European Signal Processing Conference*. Poznan, Poland: IEEE.

Shardt, Y. A. W., Brooks, K. (2018). Automated system identification in mineral processing industries: A case study using the zinc flotation cell. *IFAC-PapersOnLine*, v. 51, n. 18, p. 132–137.

Shardt, Y. A. W., Huang, B. (2013a). Data quality assessment of routine operating data for process identification. *Computers & Chemical Engineering*, v. 55, p. 19–27.

Shardt, Y. A. W., Huang, B. (2013b). Statistical properties of signal entropy for use in detecting changes in time series data. *Journal of Chemometrics*, v. 27, n. 11, p. 394–405.

Shardt, Y. A. W., Shah, S. L. (2014). Segmentation Methods for Model Identification from Historical Process Data. In *Proceedings of the 19th World Congress*. Cape Town, South Africa: IFAC, 2014. p. 2836–2841.

Verhaegen, M., Verdult, V. (2007). *Filtering and System Identification: A Least Square Approach*. Cambridge, UK: Cambridge University Press.

Wahlberg, B. System identification using laguerre models. IEEE Transactions on Automatic Control, IEEE, v. 36, n. 5, p. 551–562, 1991.

Wang, J. *et al*. (2018). Searching historical data segments for process identification in feedback control loops. *Computers and Chemical Engineering*, v. 112, n. 6, p. 6–16.