sciendo

Int. J. Appl. Math. Comput. Sci., 2024, Vol. 34, No. 1, 65–79
DOI: 10.61822/amcs-2024-0005

amcs

RECURSIVE IDENTIFICATION OF NOISY AUTOREGRESSIVE MODELS VIA A NOISE–COMPENSATED OVERDETERMINED INSTRUMENTAL VARIABLE METHOD

MATTEO BARBIERI [a], ROBERTO DIVERSI [a,*]

[a]Department of Electrical, Electronic and Information Engineering “Guglielmo Marconi” (DEI)
University of Bologna
Viale del Risorgimento 2, 40136 Bologna, Italy
e-mail: {matteo.barbieri15,roberto.diversi}@unibo.it

The aim of this paper is to develop a new recursive identification algorithm for autoregressive (AR) models corrupted by additive white noise. The proposed approach relies on a set of both low-order and high-order Yule–Walker equations and on a modified version of the overdetermined recursive instrumental variable method, leading to the estimation of both the AR coefficients and the additive noise variance. The main motivation behind our proposition is introducing model identification procedures suitable for implementation on edge-computing platforms and programmable logic controllers (PLCs), which are known to have limited capabilities and resources when dealing with complex mathematical computations (i.e., matrix inversion). Indeed, our development is focused on condition monitoring systems, with particular attention paid to their integration onboard industrial machinery. The performance of the recursive approach is tested using both numerical simulations and a laboratory case study. The obtained results are very promising.

Keywords: system identification, noisy autoregressive models, recursive estimation, Yule–Walker equations, condition monitoring systems.

1. Introduction

Autoregressive (AR) models play a very important role in the description and analysis of time series (Box *et al*., 2015) and find many applications, for instance, in spectral estimation (Kay, 1988; Stoica and Moses, 2005), speech processing (Lim and Oppenheim, 1978; Kovacevic *et al*., 1995; Grivel *et al*., 2002; Bobillet *et al*., 2007), biomedical signal processing (Pardey *et al*., 1996; Güler *et al*., 2001; Zhang *et al*., 2017), structural health monitoring (Guidorzi *et al*., 2014), fault diagnosis and condition monitoring (Basseville, 1988; Baillie and Mathew, 1996; Wang and Wong, 2002; Wang and Makis, 2009; Sikora and Sikora, 2012; Barbieri *et al*., 2019), biotechnology (Hilgert and Vila, 1999) and radar signal processing (Haykin and Steinhardt, 1992; Abramovich *et al*., 2007; Rouffet *et al*., 2015). The widespread use of this class of models is mainly due to the existence of simple and robust algorithms for their identification (e.g., least squares), the stability of the associated predictors and easy implementation of online estimation algorithms (Box *et al*., 2015).

In many practical cases, however, the signal to be described through an AR process is corrupted by measurement noise. For this reason, meticulous attention has been devoted in the literature to the problem of identifying AR models in the presence of additive white noise. In this framework, it is well known that classical identification methods like least squares (LS) and Yule–Walker (YW) equations lead to biased estimates (Kay, 1979; Zheng, 1999).

Since an AR model of order p observed in additive white noise is equivalent to an ARMA(p, p) model, the estimation of its parameters can be performed through an (overdetermined) set of the high-order Yule–Walker equations (Kay, 1980; Chan and Langford, 1982; Cadzow, 1982). It has been shown that more accurate estimates can be obtained by using both low-order and high-order Yule–Walker equations (Kay, 1980; Paliwal, 1988; Davila, 1998). However, in order to exploit also the low-order YW equations, an estimate of the additive noise

[*]Corresponding author

© 2024 M. Barbieri and R. Diversi.
This is an open access article distributed under
the Creative Commons Attribution-NonCommercial-NoDerivs license
(http://creativecommons.org/licenses/by-nc-nd/3.0/).

variance is required. The major part of the proposed algorithms for identifying noisy AR models is based on bias-compensated least squares techniques. The rationale behind these methods consists in 'compensating' the bias of classical LS through an estimate of the additive noise variance (Zheng, 1999; 2000; 2006; Jia *et al.*, 2003; Mahmoudi and Karimi, 2010; 2011; Xia and Zheng, 2015; Esfandiari *et al.*, 2020). Diversi *et al.* (2005; 2008), Petitjean *et al.* (2010) and Diversi (2018) treated the identification of noisy AR processes as an errors-in-variables (EIV) identification problem. These approaches consist in searching for the solution of the problem (AR parameters and additive and driving noise variances) inside a locus of solutions compatible with the second-order statistics of the noisy data. Other approaches rely on a state space representation of the noise-corrupted AR model (see, e.g., Grivel *et al.*, 2002; Labarre *et al.*, 2006).

As is well known, in many situations the identification must be performed online, that is, the parameter estimates must be updated as soon as new data become available (Ljung, 1999; Söderström and Stoica, 1989). Very few papers deal with the recursive identification of AR models in the presence of additive noise. The algorithms proposed by Sakai and Arase (1979) or Zheng (1997) are based on the so-called bias-compensated least squares (BCLS) technique. They consist in repeating the following two steps until convergence: (i) estimate the additive noise variance employing the current estimate of the AR parameters; (ii) estimate the AR parameters by 'compensating' the (biased) least squares solution employing the current estimate of the additive noise variance.

Despite their numerical efficiency, these methods may be affected by convergence problems and may also lead to poor results when the signal to noise ratio is not sufficiently high (Diversi *et al.*, 2008). The bias compensation principle can also be exploited to modify the least-mean-square (LMS) method to counteract the presence of the additive noise. This leads to the γ-LMS algorithm (Treichler, 1979) and the ρ-LMS algorithm (Wu and Chen, 1997). It is worth noting that these techniques require *a priori* knowledge of the additive noise variance. Moreover, convergence is achieved by using a large number of samples (e.g., a few thousand).

The β-LMS algorithm proposed by Zhang *et al.* (2000) does not require the knowledge of the additive noise variance. Nevertheless, at each iteration, a nonlinear equation must be solved, so this method can no longer be considered a recursive algorithm. Petitjean *et al.* (2009) describe an online version of the EIV approach developed by Diversi *et al.* (2008). As previously mentioned, the offline approach by Diversi *et al.* (2008) consists in solving a constrained optimization problem, that is, in minimizing a loss function inside a locus of admissible solutions. To make it recursive, Newton's method is exploited. However, the resulting identification algorithm is not 'fully' recursive as at each step it is necessary to compute the inverse of a $(p+1) \times (p+1)$ matrix, where p is the order of the AR model.

The aim of this paper is to develop a recursive identification algorithm for AR models corrupted by additive white noise without involving the inversion of matrices whose dimensions depend on the model order. The proposed approach relies on an offline algorithm that takes advantage of both the low-order and the high-order Yule–Walker equations. This leads to a bilinear system of equations involving both the AR coefficients and the additive noise variance. These equations are also known as noise-compensated Yule–Walker equations. The system can be solved in an iterative manner by computing, at each step, the solutions of two separate least-squares problems. The recursive version of the identification algorithm is obtained by modifying the overdetermined recursive instrumental variable (ORIV) method developed by Friedlander (1984) for the estimation of the autoregressive part of ARMA processes. The developed algorithm can then be called the "noise-compensated overdetermined recursive instrumental variable" (NC-ORIV). It is worth emphasizing that the NC-ORIV does not require the knowledge of the additive noise variance. Moreover, unlike the approach introduced by Petitjean *et al.* (2009), the resulting online algorithm requires only the inverse of a 2×2 matrix, which can be easily computed, so that it is 'fully' recursive. Finally, as shown in Section 5, it achieves convergence by using a small number of samples.

The main motivation behind the paper is about introducing model identification procedures suitable for implementation on edge-computing platforms and programmable logic controllers (PLCs), which are known to have limited capabilities and resources when dealing with complex mathematical computations, such as matrix inversion. In particular, our development was driven by the need to perform condition monitoring on board automatic machinery, which is controlled by means of PLCs (Barbieri *et al.*, 2018; 2021). In this context, the machinery internal condition can be monitored using accelerometers. Those sensors are mainly of two typologies; piezoelectric and MEMS-based. The former usually provide significantly better noise performance than the latter, but at higher costs. Indeed, accelerometer signals can be assumed as time series and modeled as autoregressive processes (Baillie and Mathew, 1996; Wang and Wong, 2002; Wang and Makis, 2009; Barbieri *et al.*, 2021). With the algorithm we propose, we provide a more robust way to perform condition monitoring using accelerometers, both with the cheaper and more noisy MEMS and the more expensive piezoelectric ones. We introduce an efficient and less resource intensive recursive solution to tackle the identification of noisy sensor measurements (i.e., with MEMS), while also providing a way to monitor the state of health of the sensors themselves through the estimation of the measurement noise level (for both sensor typologies). The noise variance is an indication of the sensor efficiency; an increase in such a quantity can signal the degradation of the measurement.

The performance of the recursive approach has been tested on both synthetic and real data. For the former case, the effectiveness of the recursive algorithm has been evaluated by means of Monte Carlo simulations. For the latter case, we use a laboratory setup aimed at developing condition monitoring algorithms for bearings through vibration signals. It is worth highlighting that bearing degradation modeling plays an important role in fault diagnosis of mechanical systems (Lipiec *et al.*, 2022). The obtained results are very promising.

The paper is organized as follows. Section 2 states the identification problem. Section 3 describes an offline identification algorithm that serves as a starting point for the development of the recursive algorithm proposed in Section 4. It is also shown how to exploit the statistical properties of the noisy AR process to validate the identified model and to estimate the model order. The effectiveness of the recursive approach is analyzed in Section 5 by means of simulations performed on synthetic data. A laboratory case study application is described in Section 6. Section 7 concludes the paper.

2. Problem statement

Let us consider the following p-th order AR process:

$$x(t) + a_1\, x(t-1) + \cdots + a_p\, x(t-p) = e(t), \tag{1}$$

where $e(t)$ is the driving noise. The AR signal $x(t)$ is corrupted by the additive noise $w(t)$ so that the available measurement $y(t)$ is given by

$$y(t) = x(t) + w(t). \tag{2}$$

Then, we consider the following assumptions.

A1. The AR process (1) is asymptotically stable, i.e., all roots of the polynomial

$$A(z) = z^p + a_1\, z^{p-1} + \cdots + a_{p-1}\, z + a_p \tag{3}$$

lie inside the unit disc in the z-plane.

A2. The order p of the AR model is assumed to be known *a priori*.

A3. The driving noise signal $e(t)$ is a zero-mean ergodic white process with variance σ_e^2.

A4. The additive noise $w(t)$ is a zero-mean ergodic white process with variance σ_w^2.

A5. Lastly, $e(t)$ and $w(t)$ are mutually uncorrelated.

Under these assumptions, the noisy AR identification problem can be stated as follows.

Problem 1. (*Identification problem*) Given the set of noisy output data $y(1), y(2), \ldots, y(N)$, determine a recursive estimate of the coefficients $a_1, a_2, \ldots, a_p$ and the noise variances σ_e^2, σ_w^2.

By defining the vectors

$$\varphi_x(t) = [\, -x(t-1)\ \ldots\ -x(t-p)\,]^T, \tag{4}$$

$$\varphi_y(t) = [\, -y(t-1)\ \ldots\ -y(t-p)\,]^T, \tag{5}$$

$$\varphi_w(t) = [\, -w(t-1)\ \ldots\ -w(t-p)\,]^T \tag{6}$$

and the parameter vector

$$\theta = \left[\, a_1\, a_2\, \cdots\, a_p \right]^T, \tag{7}$$

Equations (1) and (2) can be rewritten as

$$x(t) = \varphi_x^T(t)\,\theta + e(t), \tag{8}$$

$$\varphi_y(t) = \varphi_x(t) + \varphi_w(t). \tag{9}$$

The vector form of (8) and (9) will be useful for the subsequent analysis.

3. Offline identification algorithm

Given the premises of the identification of noisy AR models, we introduce the offline algorithm for identifying the model (1), (2). It will represent the starting point in developing the recursive algorithm described in Section 4. In the following, we will denote as $r_y(\tau)$ the autocorrelation at lag τ of the signal $y(t)$:

$$r_y(\tau) = E[y(t)\, y(t-\tau)] = r_y(-\tau), \tag{10}$$

where $E[\cdot]$ represents the expectation operator.

From (2) and (8) it is easy to get

$$y(t) = \varphi_x^T(t)\,\theta + e(t) + w(t); \tag{11}$$

then, by using (9), we obtain

$$y(t) = \varphi_y^T(t)\,\theta - \varphi_w^T(t)\,\theta + e(t) + w(t). \tag{12}$$

Define now the extended regressor vector

$$\begin{aligned}\bar{\varphi}_y(t) =& [-y(t-1)\ \ldots\ -y(t-p)\\ &\quad -y(t-p-1)\ \ldots\ -y(t-p-q)]^T,\end{aligned} \tag{13}$$

where q is chosen such that $q \geq p$ (see Remark 1). Multiplying both the sides of (12) by $\bar{\varphi}_y(t)$ and applying the expectation operator, we get

$$\begin{aligned}&E[\bar{\varphi}_y(t)\, y(t)]\\ &\quad = E[\bar{\varphi}_y(t)\, \varphi_y^T(t)]\,\theta - E[\bar{\varphi}_y(t)\, \varphi_w^T(t)]\,\theta\\ &\qquad + E[\bar{\varphi}_y(t)(e(t) + w(t))].\end{aligned} \tag{14}$$

Because of Assumptions A3–A5 and $\bar{\varphi}_y(t) = \bar{\varphi}_x(t) + \bar{\varphi}_w(t)$, we have

$$\begin{aligned}
&E[\bar{\varphi}_y(t)\,\varphi_w^T(t)] \\
&\quad = E[\bar{\varphi}_x(t)\,\varphi_w^T(t)] + E[\bar{\varphi}_w(t)\,\varphi_w^T(t)] \\
&\quad = E[\bar{\varphi}_w(t)\,\varphi_w^T(t)],
\end{aligned} \tag{15}$$

$$\begin{aligned}
&E[\bar{\varphi}_y(t)(e(t)+w(t))] \\
&\quad = E[\bar{\varphi}_x(t)(e(t)+w(t))] \\
&\qquad + E[\bar{\varphi}_w(t)(e(t)+w(t))] = 0.
\end{aligned} \tag{16}$$

The relations (14)–(16) lead to

$$\rho = R\,\theta - \sigma_w^2\,J\,\theta, \tag{17}$$

where

$$\begin{aligned}
R &= E[\bar{\varphi}_y(t)\,\varphi_y^T(t)] \\
&= \begin{bmatrix}
r_y(0) & r_y(1) & \cdots & r_y(p-1) \\
r_y(1) & r_y(0) & \cdots & r_y(p-2) \\
\vdots & & \ddots & \vdots \\
r_y(p-1) & r_y(p-2) & \cdots & r_y(0) \\
r_y(p) & r_y(p-1) & \cdots & r_y(1) \\
\vdots & & & \vdots \\
r_y(p+q-1) & \cdots & \cdots & r_y(q)
\end{bmatrix},
\end{aligned} \tag{18}$$

$$\begin{aligned}
\rho = E[\bar{\varphi}_y(t)\,y(t)] = -\,\big[r_y(1) \quad r_y(2) \ \cdots\ r_y(p) \\
r_y(p+1)\ \cdots\ r_y(p+q)\big]^T
\end{aligned} \tag{19}$$

and

$$J = \begin{bmatrix} I_{p\times p} \\ 0_{q\times p} \end{bmatrix}. \tag{20}$$

Since the autocorrelations $r_y(\tau), \tau = 0, \dots, p+q$ can be estimated directly from the available noisy measurements, the relation (17) can be seen as a system of $p+q$ equations in the $p+1$ unknowns $a_1, a_2, \dots, a_p$ and σ_w^2. These equations are also known as "noise-compensated Yule–Walker (NCYW)" equations. In fact, the first p equations (low-order equations) are 'corrupted' by the additive noise variance σ_w^2 whereas the last q equations (high-order equations) are noise-free.

Remark 1. As shown by Davila (2001), the condition $q \geq p$ is both necessary and sufficient to guarantee the existence of a unique solution to the NCYW equations (17). Therefore, if $q \geq p$, only the true parameter vector θ and the true additive noise variance σ_w^2 satisfy the set of equations (17).

Consider now the following partitions of R and ρ:

$$R = \begin{bmatrix} R_L \\ R_H \end{bmatrix}, \qquad \rho = \begin{bmatrix} \rho_L \\ \rho_H \end{bmatrix}, \tag{21}$$

where R_L and R_H have dimensions $p \times p$ and $q \times p$ while ρ_L and ρ_H are $p \times 1$ and $q \times 1$ column vectors. The set of equations (17) can thus be split as follows:

$$\rho_L = R_L\,\theta - \sigma_w^2\,\theta, \tag{22}$$

$$\rho_H = R_H\,\theta. \tag{23}$$

If the additive noise variance σ_w^2 were known, an estimate of θ could be directly computed from (17):

$$\hat{\theta} = \big(R - \sigma_w^2\,J\big)^{+}\rho, \tag{24}$$

where R^{+} denotes the pseudoinverse of R. Conversely, if the parameter vector θ were known, the additive noise variance could be estimated by using the first p equations of (17), i.e., by means of (22):

$$\hat{\sigma}_w^2 = \frac{\theta^T\big(R_L\,\theta - \rho_L\big)}{\theta^T\,\theta}. \tag{25}$$

Starting from (24) and (25), it is possible to devise an iterative (offline) identification algorithm for estimating θ and σ_w^2, whose steps are summarized in Algorithm 1.

Algorithm 1. Offline algorithm.

Step 1. Compute, from the available noisy data $y(1), y(2), \dots, y(N)$, the sample estimates of R and ρ:

$$\hat{R} = \frac{1}{N-p-q}\sum_{t=p+q+1}^{t=N} \bar{\varphi}_y(t)\,\varphi_y^T(t) = \begin{bmatrix} \hat{R}_L \\ \hat{R}_H \end{bmatrix},$$

$$\hat{\rho} = \frac{1}{N-p-q}\sum_{t=p+q+1}^{t=N} \bar{\varphi}_y(t)\,y(t) = \begin{bmatrix} \hat{\rho}_L \\ \hat{\rho}_H \end{bmatrix}.$$

Step 2. Determine an initial estimate $\hat{\theta}^0$ of θ and set $\hat{\theta}^k = \hat{\theta}^0$.

Step 3. Compute an estimate of the additive noise variance as follows:

$$\hat{\sigma}_w^{2k} = \frac{\hat{\theta}^{k\,T}\big(\hat{R}_L\,\hat{\theta}^k - \hat{\rho}_L\big)}{\hat{\theta}^{k\,T}\,\hat{\theta}^k}.$$

Step 4. Update the estimate of the parameter vector:

$$\hat{\theta}^{k+1} = \big(\hat{R} - \hat{\sigma}_w^{2k}\,J\big)^{+}\hat{\rho}.$$

Step 5. Set $\hat{\theta}^{k+1} = \hat{\theta}^k$ and go to Step 3.

Step 6. Repeat Steps 3–5 until

$$\frac{\|\hat{\theta}^{k+1} - \hat{\theta}^k\|}{\|\hat{\theta}^{k+1}\|} < \epsilon,$$

where ϵ is a convergence threshold.

Remark 2. (*Algorithm convergence*) It is worth noting that Algorithm 1 leads to the solution of the optimization problem

$$\min_{\theta,\sigma_w^2} f(\theta,\sigma_w^2) = \|\hat{\rho} - (\hat{R} - \sigma_w^2\, J)\,\theta\|^2. \tag{26}$$

In fact, (17) is bilinear in the unknowns θ and σ_w^2 (see (24), (25)) so that Steps 3 and 4 of Algorithm 1 compute the solutions of two separate least-squares problems. More precisely, Step 3 solves the LS problem

$$\min_{\sigma_w^2} f(\hat{\theta}^k, \sigma_w^2), \tag{27}$$

whereas Step 4 solves the LS problem

$$\min_{\theta} f(\theta, \hat{\sigma}_w^{2k}). \tag{28}$$

It follows that

$$f(\hat{\theta}^{k+1}, \hat{\sigma}_w^{2k}) \le f(\hat{\theta}^k, \hat{\sigma}_w^{2k}) \le f(\hat{\theta}^k, \hat{\sigma}_w^{2(k-1)}). \tag{29}$$

The above property guarantees the convergence of the iterative identification algorithm. In fact, it is a 'cyclic minimizer' (Stoica and Moses, 2005).

Remark 3. (*Estimation of the driving noise variance*) Once θ and σ_w^2 have been estimated, an estimate of the driving noise variance σ_e^2 can be obtained from (12). In fact, multiplying both the sides of (12) by $y(t)$ and taking the expectation, we get

$$\begin{aligned} E[y^2(t)] &= E[y(t)\,\varphi_y^T(t)]\,\theta - E[y(t)\,\varphi_w^T(t)]\,\theta \\ &\quad + E[y(t)\,e(t)] + E[y(t)\,w(t))], \end{aligned} \tag{30}$$

which, since $E[x(t)\,e(t)] = \sigma_e^2$, leads to

$$r_y(0) = \rho_L^T\,\theta + \sigma_e^2 + \sigma_w^2. \tag{31}$$

The estimate of σ_e^2 can thus be easily computed when the estimates $\hat{\theta}$ and $\hat{\sigma}_w^2$ are available:

$$\hat{\sigma}_e^2 = \hat{r}_y(0) - \hat{\rho}_L^T\,\hat{\theta} - \hat{\sigma}_w^2. \tag{32}$$

Remark 4. At each iteration of Algorithm 1, it is advisable to perform a check on the estimate $\hat{\sigma}_w^{2k}$ in order to keep it within the range $(0, \hat{r}_y(0))$. In fact, on the one hand, $\hat{\sigma}_w^{2k}$ must be a positive number while, on the other, it follows from (2) that

$$r_y(0) = r_x(0) + \sigma_w^2, \tag{33}$$

where $r_y(0) = \sigma_y^2$, $r_x(0) = \sigma_x^2$ are the variances of $y(t)$ and $x(t)$, respectively. Consequently, σ_w^2 is lower than $r_y(0)$.

Remark 5. Steps 3 and 4 of Algorithm 1 can be swapped. In this case, an initial estimate $\hat{\sigma}_w^{20}$ of the additive noise variance has to be set in Step 2. As stated in Remark 4, $\hat{\sigma}_w^{20}$ should lie in the range $(0, \hat{r}_y(0))$.

3.1. Model assessment.

An important step in system identification consists in checking the validity of the estimated model and of the assumptions behind the identification algorithm. When dealing with the noisy AR model described by Eqns. (1) and (2) a model assessment can be performed by exploiting the statistical properties of its residual. By inserting (2) into (1) we get

$$\begin{aligned} &y(t) + a_1\,y(t-1) + \cdots + a_p\,y(t-p) \\ &\quad = e(t) + w(t) + a_1\,w(t-1) + \cdots + a_p\,w(t-p). \end{aligned} \tag{34}$$

Once a model is available, the following residual can be computed:

$$\varepsilon(t) = y(t) + a_1\,y(t-1) + \cdots + a_p\,y(t-p). \tag{35}$$

From (34) it is clear that the residual is not white as it is given by the sum of the white process $e(t)$ and the moving average (MA) process $w(t) + a_1\,w(t-1) + \cdots + a_p\,w(t-p)$. Starting from Assumptions A3–A5, it is easy to show that the autocorrelation function $r_\varepsilon(\tau) = E[\varepsilon(t)\,\varepsilon(t-\tau)]$ is given by

$$\begin{aligned} &r_\varepsilon(0) = \sigma_w^2 \sum_{i=0}^{p} a_i^2 + \sigma_e^2, \\ &r_\varepsilon(\tau) = \sigma_w^2 \sum_{i=0}^{p-\tau} a_i\,a_{i+\tau}, \quad 0 < \tau \le p, \\ &r_\varepsilon(\tau) = 0, \quad \forall \tau > p. \end{aligned}$$

Therefore, $r_\varepsilon(\tau)$ behaves like the autocorrelation function of a moving average process of order p. It is thus possible to consider the normalized autocorrelation function

$$\gamma(\tau) = \frac{r_\varepsilon(\tau)}{r_\varepsilon(0)}, \quad \tau > 0, \tag{36}$$

and check its statistical properties (Box *et al.*, 2015).

In practice, we expect that the values of $\gamma(\tau)$ will be negligible for $\tau > p$. Once a model has been estimated, model assessment can be based on the following steps:

1. Compute the residual sequence $\varepsilon(1), \varepsilon(2), \dots, \varepsilon(N)$ by using (35).
2. Compute the autocorrelation samples $\hat{r}_\varepsilon(0)$, $\hat{r}_\varepsilon(1), \dots, \hat{r}_\varepsilon(M)$, where $M > p$ is a user-chosen parameter.
3. Compute the normalized autocorrelation samples $\hat{\gamma}(1), \dots, \hat{\gamma}(M)$.
4. Check if $\hat{\gamma}(\tau)$ is negligible for $\tau > p$ by using Bartlett's approximation (Box *et al.*, 2015).

It is worth noting that the above procedure can also be exploited to find an estimate of the model order p.

4. Recursive identification algorithm

In order to develop a recursive version of Algorithm 1, it is not advisable to use (24) directly as in Step 4 because of the presence of the pseudoinverse of $\hat{R} - \hat{\sigma}_w^{2k} J$. Instead, let us start from Eqn. (17) and rewrite it as

$$R\,\theta = \rho + \sigma_w^2\, J\,\theta, \tag{37}$$

from which the parameter vector θ may be obtained as

$$\theta = R^+\,\rho + \sigma_w^2\, R^+\, J\,\theta, \tag{38}$$

where R^+ is the pseudoinverse of R. In what follows, we denote by $\hat{\theta}(t)$ an estimate of θ obtained from the output samples $y(1), y(2), \dots, y(t)$. This notation will also be used for other estimated variables. Equation (38) can be exploited to compute the estimate of θ at time t given its value at time $t-1$:

$$\hat{\theta}(t) = \hat{R}^+(t)\,\hat{\rho}(t) + \hat{\sigma}_w^2(t-1)\,\hat{R}^+(t)\,J\,\hat{\theta}(t-1). \tag{39}$$

In this fashion, at each iteration we need to update $\hat{R}^+(t)$, $\hat{\rho}(t)$ and rely on the last computed value of $\hat{\sigma}_w^2$ by using Eqn. (25); note that, in the absence of measurement noise ($\sigma_w^2 = 0$), Eqn. (39) reduces to $\hat{\theta}(t) = \hat{R}^+(t)\hat{\rho}(t)$, which can be seen as an extended instrumental variable estimate of the AR model (Söderström and Stoica, 1989). This is also called an 'overdetermined instrumental variable method' by Friedlander (1984), with reference to the identification of the autoregressive part of ARMA models. Therefore, the recursive update of the quantities $\hat{R}^+(t)$ and $\hat{\rho}(t)$ can be obtained by exploiting the results developed by Friedlander (1984) (see also Söderström and Stoica, 1989). To apply that reasoning, we start by rewriting Eqn. (39), expanding $\hat{R}^+(t) = \left(\hat{R}^T(t)\hat{R}(t)\right)^{-1}\hat{R}^T(t)$ into its components:

$$\begin{aligned}\hat{\theta}(t) &= \hat{P}(t)\hat{R}^T(t)\,\hat{\rho}(t)\\ &\quad + \hat{\sigma}_w^2(t-1)\,\hat{P}(t)\hat{R}^T(t)\,J\,\hat{\theta}(t-1),\end{aligned} \tag{40}$$

where

$$\hat{P}(t) = \left(\hat{R}^T(t)\hat{R}(t)\right)^{-1}. \tag{41}$$

In the next subsection we will show how to achieve the recursive update of $\hat{P}(t)$, $\hat{R}(t)$ and $\hat{\rho}(t)$ in the absence of noise (i.e., $\sigma_w^2 = 0$). Then, the obtained results will be extended to the case of noisy autoregressive models.

4.1. Recursive algorithm: The noise-free case. The original overdetermined recursive instrumental-variable (ORIV) method (Söderström and Stoica, 1989; Friedlander, 1984) refers to the estimate

$$\hat{\theta} = \left[\sum_{s=p+q+1}^{N} \bar{\varphi}_y(s)\varphi_y^T(s)\right]^+ \left[\sum_{s=p+q+1}^{N} \bar{\varphi}_y(s)y(s)\right], \tag{42}$$

which does not involve the cross-covariance quantities R and ρ required by Algorithm 1. Instead, our starting point is the estimate

$$\begin{aligned}\hat{\theta} &= \hat{R}^+\hat{\rho}\\ &= \left[\frac{1}{N-p-q}\sum_{\tau=p+q+1}^{N} \bar{\varphi}_y(\tau)\varphi_y^T(\tau)\right]^+\\ &\quad\times\left[\frac{1}{N-p-q}\sum_{\tau=p+q+1}^{N} \bar{\varphi}_y(\tau)y(\tau)\right].\end{aligned} \tag{43}$$

To introduce a recursive version, Eqn. (43) is rewritten in the following way:

$$\hat{\theta}(t) = \hat{R}^+(t)\hat{\rho}(t) = \hat{P}(t)\hat{R}^T(t)\hat{\rho}(t), \tag{44}$$

where

$$\hat{R}(t) = \frac{1}{t-p-q}\sum_{s=p+q+1}^{t} \bar{\varphi}_y(s)\varphi_y^T(s), \tag{45}$$

$$\hat{\rho}(t) = \frac{1}{t-p-q}\sum_{s=p+q+1}^{t} \bar{\varphi}_y(s)y(s), \tag{46}$$

and $P(t)$ is defined in (41). Now, as stated by Söderström and Stoica (1989) as well as Friedlander (1984), the derivation of the recursive algorithm is obtained starting from the equation to estimate $\hat{\theta}(t)$, given its previous value $\hat{\theta}(t-1)$:

$$\begin{aligned}\hat{\theta}(t) &= \hat{\theta}(t-1)\\ &\quad + \hat{P}(t)\hat{R}^T(t)[\hat{\rho}(t) - \hat{R}(t)\hat{\theta}(t-1)].\end{aligned} \tag{47}$$

Now, we will study the recursion of the terms involved in the above equation (we will drop the hat symbol for the sake of clarity). Starting from (46) and setting $\tau = t-p-q$, we get

$$\begin{aligned}\rho(t) &= \frac{1}{\tau}\sum_{s=p+q+1}^{t} \bar{\varphi}_y(s)y(s)\\ &= \frac{1}{\tau}\left[\sum_{s=p+q+1}^{t-1} \bar{\varphi}_y(s)y(s) + \bar{\varphi}_y(t)y(t)\right]\\ &= \frac{1}{\tau}\left[\frac{\tau-1}{\tau-1}\sum_{s=p+q+1}^{t-1} \bar{\varphi}_y(s)y(s) + \bar{\varphi}_y(t)y(t)\right]\\ &= \frac{\tau-1}{\tau}\rho(t-1) + \frac{1}{\tau}\bar{\varphi}_y(t)y(t),\end{aligned} \tag{48}$$

with $t > p+q$. Analogously,

$$R(t) = \frac{\tau-1}{\tau}R(t-1) + \frac{1}{\tau}\bar{\varphi}_y(t)\varphi_y^T(t). \tag{49}$$

$P(t)$ is computed following the course of actions presented by Friedlander (1984) as well as Söderström and Stoica (1989); the starting point is $P^{-1}(t) = R^T(t)R(t)$:

$$\begin{aligned}
P^{-1}(t) &= \left[\frac{\tau-1}{\tau}R^T(t-1) + \frac{1}{\tau}\varphi_y(t)\bar{\varphi}_y^T(t)\right] \\
&\quad\times\left[\frac{\tau-1}{\tau}R(t-1) + \frac{1}{\tau}\bar{\varphi}_y(t)\varphi_y^T(t)\right] \\
&= \frac{(\tau-1)^2}{\tau^2}P^{-1}(t-1) + \frac{\tau-1}{\tau^2}\varphi_y(t)w^T(t) \\
&\quad + \frac{\tau-1}{\tau^2}w(t)\varphi_y^T(t) \\
&\quad + \frac{1}{\tau^2}\varphi_y(t)\bar{\varphi}_y^T(t)\bar{\varphi}_y(t)\varphi_y^T(t) \\
&= \frac{(\tau-1)^2}{\tau^2}P^{-1}(t-1) + \frac{1}{\tau^2}[w(t)\ \ \varphi_y(t)] \\
&\quad\times\begin{bmatrix} 0 & \tau-1 \\ \tau-1 & \bar{\varphi}_y^T(t)\bar{\varphi}_y(t)\end{bmatrix}\begin{bmatrix} w^T(t) \\ \varphi_y^T(t)\end{bmatrix} \\
&= \frac{(\tau-1)^2}{\tau^2}P^{-1}(t-1) + \frac{1}{\tau^2}\phi(t)\Lambda^{-1}(t)\phi^T(t),
\end{aligned} \tag{50}$$

where $w(t) = R^T(t-1)\bar{\varphi}_y(t)$, $\phi(t) = [w(t)\quad \varphi_y(t)]$ and

$$\Lambda^{-1}(t) = \begin{bmatrix} 0 & \tau-1 \\ \tau-1 & \bar{\varphi}_y^T(t)\bar{\varphi}_y(t)\end{bmatrix}. \tag{51}$$

Starting from (50), the derivation of $P(t)$ relies on the matrix inversion lemma (or the Woodbury identity)

$$\begin{aligned}
P(t) &= \frac{\tau^2}{(\tau-1)^2}P(t-1) - \frac{P(t-1)\phi(t)}{(\tau-1)^2} \\
&\quad\times\left[\Lambda(t) + \frac{\phi^T(t)P(t-1)\phi(t)}{(\tau-1)^2}\right]^{-1} \\
&\quad\times\frac{\tau^2}{(\tau-1)^2}\phi^T(t)P(t-1),
\end{aligned} \tag{52}$$

where, given

$$\begin{aligned}
\Lambda(t) &= \frac{1}{(\tau-1)^2}\begin{bmatrix} -\bar{\varphi}_y^T(t)\bar{\varphi}_y(t) & \tau-1 \\ \tau-1 & 0\end{bmatrix} \\
&= \frac{\bar{\Lambda}(t)}{(\tau-1)^2},
\end{aligned} \tag{53}$$

we have

$$\begin{aligned}
P(t) &= \frac{\tau^2}{(\tau-1)^2}P(t-1) - \frac{\tau^2}{(\tau-1)^2}P(t-1)\phi(t) \\
&\quad\times\left[\bar{\Lambda}(t) + \phi^T(t)P(t-1)\phi(t)\right]^{-1} \\
&\quad\times\phi^T(t)P(t-1).
\end{aligned} \tag{54}$$

At this point, the remaining part of the update of $\hat{\theta}(t)$ becomes

$$\begin{aligned}
&R^T(t)(\rho(t) - R(t)\hat{\theta}(t-1)) \\
&= \left(\frac{\tau-1}{\tau}R^T(t-1) + \frac{1}{\tau}\varphi_y(t)\bar{\varphi}_y^T(t)\right)\Bigg\{\frac{\tau-1}{\tau}\rho(t-1) \\
&\quad + \frac{\bar{\varphi}_y(t)y(t)}{\tau} - \Bigg(\frac{\tau-1}{\tau}R(t-1) \\
&\quad + \frac{\bar{\varphi}_y(t)\varphi_y^T(t)}{\tau}\Bigg)\hat{\theta}(t-1)\Bigg\} \\
&= \frac{1}{\tau^2}\left[\varphi_y(t)\quad (\tau-1)R^T(t-1)\bar{\varphi}_y(t) + \varphi_y(t)\bar{\varphi}_y^T(t)\bar{\varphi}_y(t)\right] \\
&\quad\times\begin{bmatrix} \bar{\varphi}_y^T(t)\left(\rho(t-1) - (\tau-1)R(t-1)\hat{\theta}(t-1)\right) \\ y(t) - \varphi_y^T(t)\hat{\theta}(t-1)\end{bmatrix} \\
&= \frac{1}{\tau^2}[w(t)\quad \varphi_y(t)]\begin{bmatrix} 0 & \tau-1 \\ \tau-1 & \bar{\varphi}_y^T(t)\bar{\varphi}_y(t)\end{bmatrix} \\
&\quad\times\left\{\begin{bmatrix} \bar{\varphi}_y^T(t)\rho(t-1) \\ y(t)\end{bmatrix} - \begin{bmatrix} w^T(t) \\ \varphi_y(t)\end{bmatrix}\hat{\theta}(t-1)\right\} \\
&= \frac{1}{\tau^2}\phi(t)\Lambda^{-1}(t)\left(v(t) - \phi^T(t)\hat{\theta}(t-1)\right),
\end{aligned} \tag{55}$$

where $v(t) = [\,\bar{\varphi}_y^T(t)\rho(t-1)\ y(t)\,]^T$. By substituting (55) in (47) and by exploiting the push-through identity to further develop the term $\frac{1}{\tau^2}P(t)\phi(t)$, we get

$$\begin{aligned}
&\frac{1}{\tau^2}P(t)\phi(t) \\
&\quad = P(t-1)\phi(t) \\
&\quad\quad \times\left[\bar{\Lambda}(t) + \phi^T(t)P(t-1)\phi(t)\right]^{-1}\Lambda(t).
\end{aligned} \tag{56}$$

The above computations lead to the time-weighted ORIV (T-ORIV) algorithm reported in Algorithm 2. The initial step may be defined in the following way:

$$\begin{aligned}
\hat{\theta}(0) &= 0, & P(0) &= \psi I, \\
\rho(0) &= 0, & R(0) &= 0,
\end{aligned} \tag{57}$$

with ψ being any large positive number.

4.2. Recursive algorithm for noisy autoregressive models.

Starting from the procedure described in Section 4.1, it is possible to obtain a recursive version of Algorithm 1. Since $P(t)$, $R(t)$ and $\rho(t)$ in (40) are the same matrices derived for the time-weighted ORIV, the recursive noise-compensated algorithm based on the overdetermined instrumental variable method follows easily from Algorithm 2 and is described in Algorithm 3. Note that, in Step 10, $R_{11}(t)$ is the top-left element of $R(t)$, which is an estimate of the output noise variance $E[y^2(t)] = r_y(0)$. The initial step of Algorithm 3 may be defined in the following way:

$$\begin{aligned}
\hat{\theta}(0) &= \alpha\mathbf{1}, & P(0) &= \psi I, \\
\rho(0) &= \beta\mathbf{1}, & R(0) &= \gamma J,
\end{aligned} \tag{58}$$

with α, β and γ being any small positive number and ψ a large positive one.

Algorithm 2. Time-weighted ORIV (T-ORIV).

```
1. $w(t) = R^T(t-1)\bar{\varphi}_y(t)$
2. $\phi(t) = [w(t) \quad \varphi_y(t)]$
3. $\bar{\Lambda}(t) = \begin{bmatrix} -\bar{\varphi}_y^T(t)\bar{\varphi}_y(t) & \tau-1 \\ \tau-1 & 0 \end{bmatrix}$
4. $v(t) = \begin{bmatrix} \bar{\varphi}_y^T(t)\rho(t-1) \\ y(t) \end{bmatrix}$
5. $K(t) = P(t-1)\phi(t)\left[\bar{\Lambda}(t) + \phi^T(t)P(t-1)\phi(t)\right]^{-1}$
6. $\hat{\theta}(t) = \hat{\theta}(t-1) + K(t)\left(v(t) - \phi^T(t)\hat{\theta}(t-1)\right)$
7. $R(t) = \frac{\tau-1}{\tau}R(t-1) + \frac{1}{\tau}\bar{\varphi}_y(t)\varphi_y^T(t)$
8. $\rho(t) = \frac{\tau-1}{\tau}\rho(t-1) + \frac{1}{\tau}\bar{\varphi}_y(t)y(t)$
9. $P(t) = \frac{\tau^2}{(\tau-1)^2}\left(P(t-1) - K(t)\phi^T(t)P(t-1)\right)$
```

Algorithm 3. Noise-compensated ORIV (NC-ORIV).

```
1. $w(t) = R^T(t-1)\bar{\varphi}_y(t)$
2. $\phi(t) = [w(t) \quad \varphi_y(t)]$
3. $\bar{\Lambda}(t) = \begin{bmatrix} -\bar{\varphi}_y^T(t)\bar{\varphi}_y(t) & \tau-1 \\ \tau-1 & 0 \end{bmatrix}$
4. $K(t) = P(t-1)\phi(t)\left[\bar{\Lambda}(t) + \phi^T(t)P(t-1)\phi(t)\right]^{-1}$
5. $R(t) = \frac{\tau-1}{\tau}R(t-1) + \frac{1}{\tau}\bar{\varphi}_y(t)\varphi_y^T(t)$
6. $\rho(t) = \frac{\tau-1}{\tau}\rho(t-1) + \frac{1}{\tau}\bar{\varphi}_y(t)y(t)$
7. $P(t) = \frac{\tau^2}{(\tau-1)^2}\left[P(t-1) - K(t)\phi^T(t)P(t-1)\right]$
8. $\hat{\theta}(t) = P(t)R^T(t)\rho(t) + \sigma_w^2(t-1)P(t)R^T(t)J\hat{\theta}(t-1)$
9. $\hat{\sigma}_w^2(t) = \frac{\hat{\theta}(t)^T\left(R_L(t)\hat{\theta}(t) - \rho_L(t)\right)}{\hat{\theta}(t)^T\hat{\theta}(t)}$
10. $\hat{\sigma}_e^2(t) = R_{11}(t) - \rho_L^T(t)\,\hat{\theta}(t) - \hat{\sigma}_w^2(t)$
```

The implementation of Algorithm 3 is indeed a bit more complex than that of the standard recursive least squares (RLS) and requires a few more computations than the standard ORIV. Here are some remarks to be taken into account when implementing the algorithm.

Remark 6. The algorithm does not need any matrix inversion of dimension $p \times p$; however, it requires the inverse of a 2×2 matrix at Step 4, which can be easily tackled during algorithm implementation.

Remark 7. To ensure numerical consistency, the algorithm implementation should monitor the value of $\hat{\sigma}_w^2$ in order to keep it at least nonnegative. Numerical errors introduced by the deployment of the algorithm may occur, in particular when $\hat{\sigma}_w^2$ is close to zero.

4.3. Forgetting factor. Recursive identification algorithms are usually employed in control identification to track system parameters evolution or in calibration and auto-commissioning procedures during system initialization to estimate relevant system parameters. For instance, electric drives perform such routines to identify motor resistance and inductance when configured for the first time. In other cases, these algorithms may be used for condition monitoring and diagnostics, exploiting the growing edge-computing power of machinery controllers. In this fashion, the estimation of parameters and noise variances continuously in time may be required when tracking variations of those quantities during system operation. However, the recursive formulation of the algorithm we developed in the previous sections does not permit real-time parameter tracking. To do this, we need a version of the algorithm that includes a forgetting factor.

The easiest way to introduce the forgetting factor is to replace the time-weighting part of Algorithm 3 with a suitable coefficient,

$$\lambda = \frac{\tau - 1}{\tau}.$$

For instance, the update of $R(t), \rho(t), P(t)$ becomes

$$R(t) = \lambda R(t-1) + (1-\lambda)\bar{\varphi}_y(t)\varphi_y^T(t), \tag{59}$$

$$\rho(t) = \lambda\rho(t-1) + (1-\lambda)\bar{\varphi}_y(t)y(t), \tag{60}$$

$$P(t) = \frac{1}{\lambda^2}\left[P(t-1) - K(t)\phi^T(t)P(t-1)\right], \tag{61}$$

with $0.95 \leq \lambda < 1$. Unlike in standard formulations where no noisy measurements are involved, we have to take into account the estimate of $\sigma_w^2(t)$ when adding the forgetting factor. This is done by keeping the estimates of $R(t), \rho(t), P(t)$ in line with their counterpart obtained through the expectation operator by using $(1-\lambda)$ in the update term. This operation is analogous to setting τ to a fixed time window, thus fixing the weights between the past samples and the new sample in the update. Finally, following the reasoning unfolded previously, the recursive algorithm with a forgetting factor is reported in Algorithm 4. The initial step may be defined in the following way:

$$\begin{aligned} \hat{\theta}(0) &= \alpha\mathbf{1}, \quad P(0) = \psi I, \\ \rho(0) &= \beta\mathbf{1}, \quad R(0) = \gamma J, \end{aligned} \tag{62}$$

with α, β and γ being any small positive numbers and ψ a large positive one.

5. Simulation results

The performance of the proposed recursive approach has been tested by means of Monte Carlo simulations and compared with that of the offline method described in

Table 1. True and estimated values of the model coefficients and of the output noise variance: Monte Carlo simulation of 1000 runs performed with $N = 5000$ and $q = 10$.

	a_1	a_2	a_3	a_4	σ_w^2
		Model (63)			SNR = 10 dB
True	-2.1690	2.8227	-2.0408	0.8853	3.6
Algorithm 1	-2.1674 ± 0.0087	2.8191 ± 0.0161	-2.0369 ± 0.0157	0.8836 ± 0.0081	3.5986 ± 0.0970
Algorithm 3	-2.1674 ± 0.0087	2.8191 ± 0.0160	-2.0372 ± 0.0153	0.8837 ± 0.0079	3.5980 ± 0.0971
		Model (64)			SNR = 10 dB
True	-1.6771	1.6875	-0.9433	0.3164	0.6
Algorithm 1	-1.6778 ± 0.0472	1.6883 ± 0.0827	-0.9439 ± 0.0727	0.3167 ± 0.0294	0.5988 ± 0.0229
Algorithm 3	-1.6780 ± 0.0510	1.6884 ± 0.0900	-0.9441 ± 0.0791	0.3167 ± 0.0318	0.5989 ± 0.0230

Algorithm 4. NC-ORIV with a forgetting factor.

```
1. $w(t) = R^T(t-1)\bar{\varphi}_y(t)$
2. $\phi(t) = [w(t) \quad \varphi_y(t)]$
3. $\bar{\Lambda}(t) = \begin{bmatrix} -\bar{\varphi}_y^T(t)\bar{\varphi}_y(t) & \frac{\lambda}{1-\lambda} \\ \frac{\lambda}{1-\lambda} & 0 \end{bmatrix}$
4. $K(t) = P(t-1)\phi(t)\left[\bar{\Lambda}(t) + \phi^T(t)P(t-1)\phi(t)\right]^{-1}$
5. $R(t) = \lambda R(t-1) + (1-\lambda)\bar{\varphi}_y(t)\varphi_y^T(t)$
6. $\rho(t) = \lambda\rho(t-1) + (1-\lambda)\bar{\varphi}_y(t)y(t)$
7. $P(t) = \frac{1}{\lambda^2}\left(P(t-1) - K(t)\phi^T(t)P(t-1)\right)$
8. $\hat{\theta}(t) = P(t)R^T(t)\rho(t) + \sigma_w^2(t-1)P(t)R^T(t)J\hat{\theta}(t-1)$
9. $\hat{\sigma}_w^2(t) = \frac{\hat{\theta}(t)^T\left(R_L(t)\hat{\theta}(t) - \rho_L(t)\right)}{\hat{\theta}(t)^T\hat{\theta}(t)}$
10. $\hat{\sigma}_e^2 = R_{11}(t) - \rho_L^T(t)\,\hat{\theta}(t) - \hat{\sigma}_w^2(t)$
```

Section 3. The following two AR processes have been considered (Diversi *et al.*, 2008):

$$\begin{aligned} &x(t) - 2.1690\,x(t-1) + 2.8227\,x(t-2) \\ &\quad - 2.0408\,x(t-3) + 0.8853\,x(t-4) = e(t), \quad (63) \\ &x(t) - 1.6771\,x(t-1) + 1.6875\,x(t-2) \\ &\quad - 0.9433\,x(t-3) + 0.3164\,x(t-4) = e(t). \quad (64) \end{aligned}$$

The model (63) represents a narrowband process with two sharp peaks which are quite close, whereas the model (64) represents a broadband process with a smooth spectrum. For both the above models, the driving process is white noise with the unit variance. Monte Carlo simulations of 1000 runs have been carried out by considering additive noise sequences with variances $\sigma_w^2 = 3.6$ for the model (63) and $\sigma_w^2 = 0.6$ for the model (64), corresponding to a signal-to-noise ratio (SNR) of 10 dB per model. The SNR is defined as

$$\mathrm{SNR} = 20\,\log_{10}\sqrt{\frac{E[x^2(t)]}{\sigma_w^2}}(\mathrm{dB}). \tag{65}$$

The number of available noisy output samples is $N = 5000$. For both the identification approaches (Algorithms 1 and 3), the parameter q (i.e., the number of high-order YW equations in (23)) has been set to $q = 10$. The obtained results are summarized in Table 1, which reports, for each experimental case, the obtained mean value of the parameters and noise variance estimates along with the associated standard deviations. The results show remarkable accuracy in estimation for both the algorithms and, in particular, suggest that the recursive approach (Algorithm 3) exhibits performances on par with its batch counterpart of Algorithm 1. This is valuable from an algorithm exploitation and deployment perspective, since Algorithm 3 brings in all the benefits of recursive algorithms in terms of computational resources.

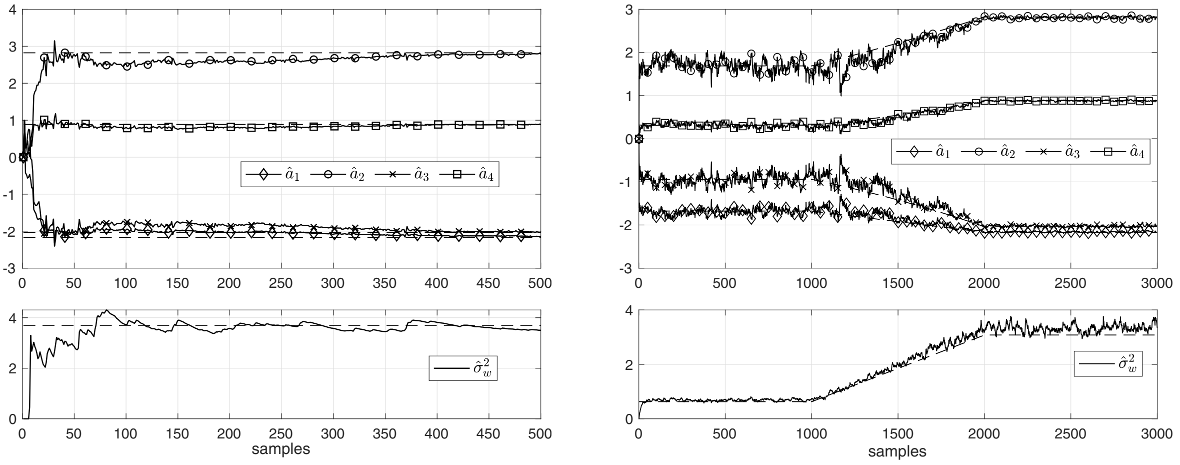
In Fig. 1(a) we show the time evolution of the quantities involved in the solution of Problem 1 applied to the identification of the model (63), i.e., $\hat{\theta}$ and $\hat{\sigma}_w^2$.

A second simulation scenario has been considered to test the tracking ability of Algorithm 4 (NC-ORIV with a forgetting factor). A sequence of $N = 3000$ samples has been generated as follows. The first 1000 samples have been generated according to the model (64). Then, the AR coefficients and the additive noise variance change linearly from the values of the model (64) to those of the model (63) (data samples from 1000 to 2000). The last 1000 samples have been generated according to the model (63). The AR coefficients and the additive noise variance have been identified by means of Algorithm 4 with a forgetting factor $\lambda = 0.999$. The results of this simulation are reported in Fig. 1(b), showing the tracking capabilities of Algorithm 4.

6. Case study application

In this section, we test the proposed algorithm in a laboratory case study application. In particular, we use a setup aimed at developing algorithms for condition monitoring of bearings on commercial hardware: programmable logic controllers (PLCs).

The main concept behind the use of recursive estimation algorithms in prognostics of automatic machines is that meaningful signals coming from the



(a)

(b)

Fig. 1. Simulation results: estimation of the coefficients and noise variance of the model (63) with the NC-ORIV (Algorithm 3) under an SNR of 10 dB (a), estimation of the coefficients and noise variance with a varying underlying model structure from the model (64) to the model (63) with the NC-ORIV under an SNR of 10 dB and $\lambda = 0.999$ (Algorithm 4) (b). The dashed lines correspond to true values.

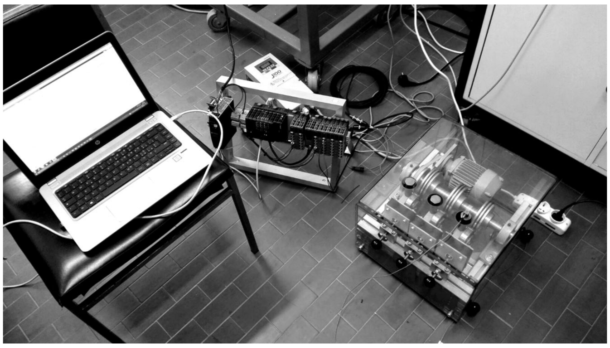


Fig. 2. Condition monitoring setup. From left to right: a PC supervisor with Automation Studio, a B&R PLC with accelerometer acquisition, and a bearing mechanism frame.

machinery can be modeled and estimated using system identification theory rules and guidelines, and we call this method the model-of-signals (MoS) (Barbieri *et al.*, 2018). In this way we can use industrial computers as edge-computing devices to retrieve valuable information regarding the machinery state of health. In particular, vibration signals can be successfully modeled by means of autoregressive processes (Baillie and Mathew, 1996; Wang and Wong, 2002; Wang and Makis, 2009; Barbieri *et al.*, 2019).

PLCs process information under real time constraints, their primary function being logic control. Complex matrices operations, such as inverses, are resource demanding in this kind of environment and the available computational power and memory are limited. However, recursive versions of identification algorithms that avoid inverse matrix computations require only basic matrix operations. In our previous work (Barbieri *et al.*, 2018), the implementation of the condition monitoring library for industrial computers laid the foundations for implementing the recursive least-squares (RLS) algorithm. Then, the library became wider and more advanced, introducing prognostics in the work of Barbieri *et al.* (2021) where the ORIV algorithm was implemented and tested with complementary decision-making algorithms, and fully integrated within the automation pyramid. In this work, we deployed effectively “on edge” Algorithm 3 and tested it in a laboratory case study application.

The monitoring architecture is shown in Fig. 2, where we supervise the processing on the PLC with a laptop that uses B&R Integrated Development Environment, Automation Studio. Then, the PLC is a X20 CP1586 model from B&R that is sampling at 5 kHz with the module X20CM4810-C01, a piezoelectric accelerometer, 356B21 from PCB, mounted on the frame of the bearing condition monitoring setup. This setup has three bearings at different wearing conditions, from new to almost broken, with one in between.

In particular, we focus on the healthy bearing to test the proposed estimation algorithm under semi-synthetic conditions. This means that we use the aforementioned setup with a piezoelectric sensor, which is known to have low measurement noise, and we artificially inject measurement noise into the system. This is done via software by using a random number generating function within the PLC program, before passing the vibration

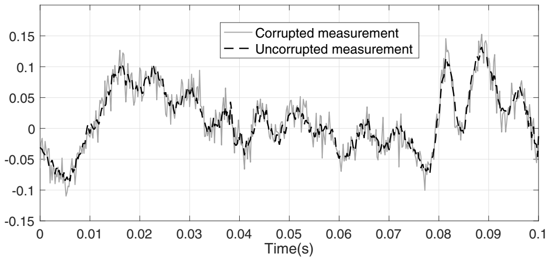


Fig. 3. Accelerometer signal: 0.1 s portion of the signal measured from the setup with and without additive noise.

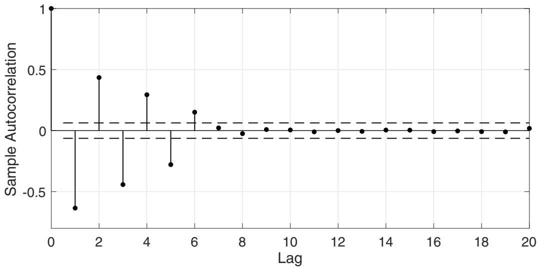


Fig. 4. Residual autocorrelation decay on the corrupted accelerometer measurements collected for the model order assessment when using $p = 6$.

samples to the estimation function block. Thus, we can control the signal to noise ratio of the injected disturbance. The noise variance estimation introduced by Algorithm 3 permits not only to exploit the algorithm in the standard MoS fashion, but also to monitor the measurement noise, allowing diagnosing and possibly prognosing also the sensor state of health, which has been a limit of our implementations until now.

The test is performed by running the bearing setup at a constant speed while the accelerometer is sampling vibrations on the healthy bearing. The PLC executes two functions, one that handles the signal buffering, storing 10 s of data (i.e., 50000 samples), and one processing those data with both Algorithms 2 and 3. The buffering function is appended to a high priority task, having a period of 0.4 ms. In this way, the program stores two samples per period. Then, once the buffer is full, the second program receives the samples, adds the measurement noise to the input to Algorithm 3, and subsequently executes the identifications algorithms. In particular, this task computes the T-ORIV estimates with uncorrupted and corrupted output and the NC-ORIV estimates with corrupted output. This procedure is appended to a low-priority task having a 10 s period. It estimates the model from the previous 10 s of measurements while the current buffer of samples is getting collected. As explained by Barbieri *et al.* (2021), the proposed scheduling of the programs on the PLC, despite losing actual real-time tracking capabilities, still allows monitoring the condition of mechanisms on-line while considerably less impacting the device's resources. Finally, the computed model of signals is sent via OPC-UA, as done by Barbieri *et al.* (2018), to a running MATLAB program collecting them every 10 s. In Fig. 3 a measurement of 0.1 s is presented to show both the uncorrupted measurement and the corrupted one.

The purpose of the test here is to show that both Algorithms 2 and 3, the former performed on noise-free measurements, the latter on noise-corrupted ones, provide similar sets of estimated parameters. In addition, we also show that Algorithm 2 produces a different model under corrupted measurements. Before starting the test we need to select a suitable model order for the estimation problem. Following the steps described in Section 3.1, we logged 50000 corrupted accelerometer samples when the system was running. Then, we used this set of measurements to compute the residual autocorrelations to check if, among a set of model orders (from 1 to 20), we were able to find a order-matching autocorrelation decay using Barlett's approximation. As shown in Fig. 4, we found $p = 6$ as a suitable model order for our experiment. The number of high-order YW equations in (23) has been set to $q = 12$.

To show the performance of the proposed solution the models estimated by applying Algorithm 2 (T-ORIV) to uncorrupted data are compared with the models identified by applying both Algorithms 2 and 3 (NC-ORIV) to corrupted data. Figure 5 reports the AR parameters collected over time for Algorithm 2 with noise-free and noisy data, whereas Fig. 6 reports the AR parameters collected over time for Algorithm 2 with noise-free data and Algorithm 3 with noisy data. As shown in Fig. 6, although the models have been estimated on real data, the NC-ORIV with noisy data and the T-ORIV with uncorrupted data provide approximately the same set of models; this is not the case for the models identified by using Algorithm 2 with noise-free and noisy data, see Fig. 5. Figure 7 shows that the additive noise variances estimated over time are very close to the true value.

To further validate our statement, we made use of the normalized root mean square error, which measures the distance between two given estimations: the closer it is to zero, the closer are the models. In particular, the following two NRMSEs are considered:

$$\mathrm{NRMSE}_{\text{Tcor-Tunc}} = \sqrt{\frac{\|\hat{\theta}_{\text{Tcor}} - \hat{\theta}_{\text{Tunc}}\|}{\|\hat{\theta}_{\text{Tunc}}\|}}, \tag{66}$$

$$\mathrm{NRMSE}_{\text{NCcor-Tunc}} = \sqrt{\frac{\|\hat{\theta}_{\text{NCcor}} - \hat{\theta}_{\text{Tunc}}\|}{\|\hat{\theta}_{\text{Tunc}}\|}}. \tag{67}$$

In the above expressions, $\hat{\theta}_{\text{Tunc}}$ denotes the AR model estimated by applying the T-ORIV to uncorrupted data,

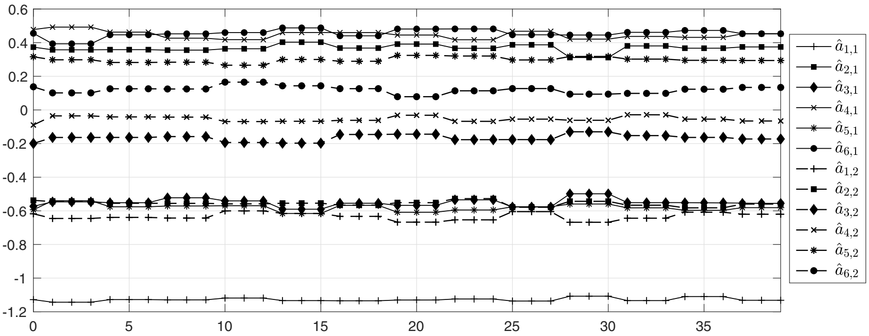


Fig. 5. Collected AR model parameters over time: $\hat{a}_{i,1}$, $i = 1, \dots, 6$ (solid lines) are computed by means of Algorithm 2 using directly the uncorrupted accelerometer signal, $\hat{a}_{i,2}$, $i = 1, \dots, 6$ (dashed lines) are computed by means of Algorithm 2 using the corrupted accelerometer signal.

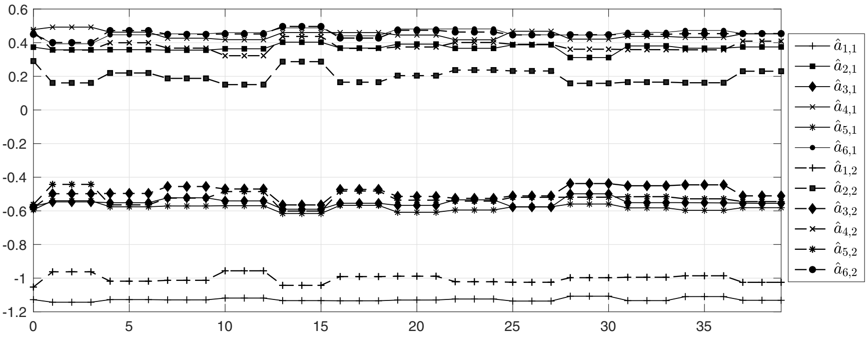


Fig. 6. Collected AR model parameters over time: $\hat{a}_{i,1}$, $i = 1, \dots, 6$ (solid lines) are computed by means of Algorithm 2 using directly the uncorrupted accelerometer signal, $\hat{a}_{i,2}$, $i = 1, \dots, 6$ (dashed lines) are computed by means of Algorithm 3 using the corrupted accelerometer signal.

which can be considered the reference model, $\hat{\theta}_{\text{Tcor}}$ is the AR model estimated by using the T-ORIV with noise-corrupted data, whereas $\hat{\theta}_{\text{NCcor}}$ is the AR model estimated by means of the NC-ORIV with noise-corrupted data. The results are shown in Fig. 8, where the estimates of Algorithm 3 (NC-ORIV) are shown to be closer to those obtained with Algorithm 2 (T-ORIV) applied to uncorrupted data, with an overall NRMSE $< 10\%$. On the other hand, T-ORIV estimates under corrupting noise are far apart from the reference ones (NRMSE $> 100\%$).

7. Conclusion

A new recursive algorithm for estimating autoregressive models in the presence of additive noise was devised. The proposed approach takes advantage, on the one hand, of a set of low-order and high-order Yule–Walker equations, on the other and, of a modified version of the overdetermined recursive instrumental variable method. The obtained recursive algorithm is able to update, at each time, both the AR coefficients and the additive noise variance, and requires only the inverse of a 2 × 2 matrix, which can be easily computed.

The presented case study showed how such an algorithm can be deployed directly into industrial machines computers (PLCs) for condition monitoring. Indeed, we exploited the algorithm's low requirements in computational resources to show that it can be used to monitor the working condition of bearings using vibration signals. Further developments include the extension of the proposed algorithm to other model structures, such as ARX models, and the integration of Algorithm 3 with machine learning algorithms to perform fault detection and isolation, not only regarding the monitored machinery, but also the monitoring sensor.

References

Abramovich, Y.I., Spencer, N.K. and Turley, M.D. (2007). Time-varying autoregressive (TVAR) models for multiple radar observations, *IEEE Transactions on Signal Processing* **55**(4): 1298–1311.

Baillie, D.C. and Mathew, J. (1996). A comparison of autoregressive modeling techniques for fault diagnosis of

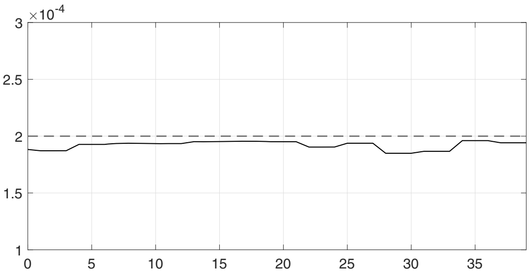


Fig. 7. Estimated additive noise variance over time by applying Algorithm 3 (NC-ORIV) to corrupted data (solid line). The dashed line refers to the true value.

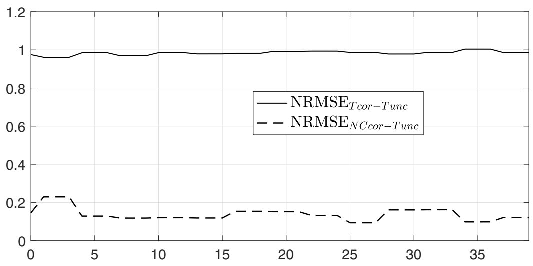


Fig. 8. Comparison of the estimated models using the NRMSE at each collection time: $\hat{\theta}_{\rm Tcor}$ vs $\hat{\theta}_{\rm Tunc}$ (solid line), $\hat{\theta}_{\rm NCcor}$ vs $\hat{\theta}_{\rm Tunc}$ (dashed line); see Eqns. (66) and (67).

rolling element bearings, *Mechanical Systems and Signal Processing* **10**(1): 1–17.

Barbieri, M., Bosso, A., Conficoni, C., Diversi, R., Sartini, M. and Tilli, A. (2018). An onboard model-of-signals approach for condition monitoring in automatic machines, *in* M. Zelm *et al.* (Eds), *Enterprise Interoperability: Smart Services and Business Impact of Enterprise Interoperability*, Wiley Online Library, London, pp. 263–269.

Barbieri, M., Diversi, R. and Tilli, A. (2019). Condition monitoring of ball bearings using estimated AR models as logistic regression features, *Proceedings of the 18th European Control Conference (ECC 2019), Naples, Italy,* pp. 3904–3909.

Barbieri, M., Nguyen, K.T., Diversi, R., Medjaher, K. and Tilli, A. (2021). RUL prediction for automatic machines: A mixed edge-cloud solution based on model-of-signals and particle filtering techniques, *Journal of Intelligent Manufacturing* **32**(5): 1421–1440.

Basseville, M. (1988). Detecting changes in signals and systems—A survey, *Automatica* **24**(3): 309–326.

Bobillet, W., Diversi, R., Grivel, E., Guidorzi, R., Najim, M. and Soverini, U. (2007). Speech enhancement combining optimal smoothing and errors-in-variables identification of noisy AR processes, *IEEE Transactions on Signal Processing* **55**(12): 5564–5578.

Box, G.E., Jenkins, G.M., Reinsel, G.C. and Ljung, G.M. (2015). *Time Series Analysis: Forecasting and Control*, John Wiley & Sons, Hoboken.

Cadzow, J. (1982). Spectral estimation: An overdetermined rational model equation approach, *Proceedings of the IEEE* **70**(9): 907–939.

Chan, Y. and Langford, R. (1982). Spectral estimation via the high-order Yule–Walker equations, *IEEE Transactions on Acoustics, Speech, and Signal Processing* **30**(5): 689–698.

Davila, C.E. (1998). A subspace approach to estimation of autoregressive parameters from noisy measurements, *IEEE Transactions on Signal Processing* **46**(2): 531–534.

Davila, C.E. (2001). On the noise-compensated Yule–Walker equations, *IEEE Transactions on Signal Processing* **49**(6): 1119–1121.

Diversi, R. (2018). Identification of multichannel AR models with additive noise: A Frisch scheme approach, *Proceedings of the 26th European Signal Processing Conference (EUSIPCO 2018), Rome, Italy,* pp. 1252–1256.

Diversi, R., Guidorzi, R. and Soverini, U. (2008). Identification of autoregressive models in the presence of additive noise, *International Journal of Adaptive Control and Signal Processing* **22**(5): 465–481.

Diversi, R., Soverini, U. and Guidorzi, R. (2005). A new estimation approach for AR models in presence of noise, *IFAC Proceedings Volumes* **38**(1): 160–165.

Esfandiari, M., Vorobyov, S.A. and Karimi, M. (2020). New estimation methods for autoregressive process in the presence of white observation noise, *Signal Processing* **171**(6): 107480.

Friedlander, B. (1984). The overdetermined recursive instrumental variable method, *IEEE Transactions on Automatic Control* **29**(4): 353–356.

Grivel, E., Gabrea, M. and Najim, M. (2002). Speech enhancement as a realisation issue, *Signal Processing* **44**(12): 1963–1978.

Guidorzi, R., Diversi, R., Vincenzi, L., Mazzotti, C. and Simioli, V. (2014). Structural monitoring of a tower by means of MEMS-based sensing and enhanced autoregressive models, *European Journal of Control* **20**(1): 4–13.

Güler, I., Kiymik, M.K., Akin, M. and Alkan, A. (2001). AR spectral analysis of EEG signals by using maximum likelihood estimation, *Computers in Biology and Medicine* **31**(6): 441–450.

Haykin, S.S. and Steinhardt, A.O. (1992). *Adaptive Radar Detection and Estimation*, Wiley-Interscience, New York.

Hilgert, N. and Vila, J.-P. (1999). D-step ahead Kalman predictor for controlled autoregressive processes with random coefficients, *International Journal of Applied Mathematics and Computer Science* **9**(1): 207–217.

Jia, L.J., Kanae, S., Yang, Z.J. and Wada, K. (2003). On bias compensation estimation for noisy AR process, *Proceedings of the 42nd IEEE Conference on Decision and Control (CDC 2003), Maui, USA,* pp. 405–410.

Kay, S. (1979). The effects of noise on the autoregressive spectral estimator, *IEEE Transactions on Acoustics, Speech, and Signal Processing* **27**(5): 478–485.

Kay, S. (1980). Noise compensation for autoregressive spectral estimates, *IEEE Transactions on Acoustics, Speech, and Signal Processing* **28**(3): 292–303.

Kay, S.M. (1988). *Modern Spectral Estimation*, Prentice-Hall, Englewood Cliffs.

Kovacevic, B.D., Milosavljevic, M.M. and Veinovic, M.D. (1995). Robust recursive AR speech analysis, *Signal Processing* **44**(6): 125–138.

Labarre, D., Grivel, E., Berthoumieu, Y., Todini, E. and Najim, M. (2006). Consistent estimation of autoregressive parameters from noisy observations based on two interacting Kalman filters, *Signal Processing* **86**(10): 2863–2876.

Lim, J. and Oppenheim, A. (1978). All-pole modeling of degraded speech, *IEEE Transactions on Acoustics, Speech, and Signal Processing* **26**(3): 197–210.

Lipiec, B., Mrugalski, M., Witczak, M. and Stetter, R. (2022). Towards a health-aware fault tolerant control of complex systems: A vehicle fleet case, *International Journal of Applied Mathematics and Computer Science* **32**(4): 619–634, DOI: 10.34768/amcs-2022-0043.

Ljung, L. (1999). *System Identification—Theory for the User*, Prentice-Hall, Upper Saddle River.

Mahmoudi, A. and Karimi, M. (2010). Parameter estimation of autoregressive signals from observations corrupted with colored noise, *Signal Processing* **90**(1): 157–164.

Mahmoudi, A. and Karimi, M. (2011). Inverse filtering based method for estimation of noisy autoregressive signals, *Signal Processing* **91**(7): 1659–1664.

Paliwal, K. (1988). A noise-compensated long correlation matching method for AR spectral estimation of noisy signals, *Signal Processing* **15**(4): 437–440.

Pardey, J., Roberts, S.K. and Tarassenko, L. (1996). A review of parametric modelling techniques for EEG analysis, *Medical Engineering & Physics* **18**(1): 2–11.

Petitjean, J., Diversi, R., Grivel, E., Guidorzi, R. and Roussilhe, P. (2009). Recursive errors-in-variables approach for AR parameter estimation from noisy observations. Application to radar sea clutter rejection, *Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2009), Taipei, Taiwan*, pp. 3401–3404.

Petitjean, J., Grivel, E., Bobillet, W. and Roussilhe, P. (2010). Multichannel AR parameter estimation from noisy observations as an errors-in-variables issue, *Signal, Image and Video Processing* **4**: 209–220.

Rouffet, T., Grivel, E., Vallet, P., Enderli, C. and Kemkemian, S. (2015). Combining two phase codes to extend the radar unambiguous range and get a trade-off in terms of performance for any clutter, *Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2015), Brisbane, Australia*, pp. 1822–1826.

Sakai, H. and Arase, M. (1979). Recursive parameter estimation of an autoregressive process disturbed by white noise, *International Journal of Control* **30**(6): 949–966.

Sikora, M. and Sikora, B. (2012). Improving prediction models applied in systems monitoring natural hazards and machinery, *International Journal of Applied Mathematics and Computer Science* **22**(2): 477–491, DOI: 10.2478/v10006-012-0036-3.

Söderström, T. and Stoica, P. (1989). *System Identification*, Prentice-Hall, Inc., Hemel Hempstead.

Stoica, P. and Moses, R. (2005). *Spectral Analysis of Signals*, Pearson Prentice Hall, Upper Saddle River.

Treichler, J.R. (1979). Transient and convergent behaviour of the adaptive line enhancer, *IEEE Transactions on Acoustics, Speech and Signal Processing* **27**(2): 53–62.

Wang, W. and Wong, A.K. (2002). Autoregressive model-based gear fault diagnosis, *Journal of Vibration and Acoustics* **124**(3): 172–179.

Wang, X. and Makis, V. (2009). Autoregressive model-based gear shaft fault diagnosis using the Kolmogorov–Smirnov test, *Journal of Sound and Vibration* **327**(3): 413–423.

Wu, W.R. and Chen, P.C. (1997). Adaptive AR modelling in white Gaussian noise, *IEEE Transactions on Signal Processing* **45**(5): 1184–1191.

Xia, Y. and Zheng, W.X. (2015). Novel parameter estimation of autoregressive signals in the presence of noise, *Automatica* **62**(12): 98–105.

Zhang, Y., Liu, B., Ji, X. and Huang, D. (2017). Classification of EEG signals based on autoregressive model and wavelet packet decomposition, *Neural Processing Letters* **45**: 365–378.

Zhang, Y., Wen, C. and Chai Soh, Y. (2000). Unbiased LMS filtering in the presence of white measurement noise with unknown power, *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* **47**(9): 968–972.

Zheng, W.X. (1997). Estimation of autoregressive signals from noisy measurements, *IEE Proceedings—Vision, Image and Signal Processing* **144**(1): 39–45.

Zheng, W.X. (1999). A least-squares based method for autoregressive signals in the presence of noise, *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* **46**(1): 81–85.

Zheng, W.X. (2000). Autoregressive parameter estimation from noisy data, *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* **47**(1): 71–75.

Zheng, W.X. (2006). On estimation of autoregressive signals in the presence of noise, *IEEE Transactions on Circuits and Systems II: Express Briefs* **53**(12): 1471–1475.

Matteo Barbieri holds an MS degree in automation engineering (2017) and a PhD in biomedical, electrical, and system engineering (2021) from the University of Bologna, Italy. His doctoral research focused on predictive maintenance for industrial machinery. Currently, as an R&D engineer at Rebernig Supervisioni, he specializes in machine learning, smart cities, energy monitoring, and predictive maintenance.

Roberto Diversi received his Laurea degree (summa cum laude) in electronic engineering and his PhD degree in system engineering from the University of Bologna, Italy, in 1996 and 2000, respectively. He is currently an associate professor with the Department of Electrical, Electronic and Information Engineering "Guglielmo Marconi" (DEI), University of Bologna. His research interests include system identification theory and applications, diagnosis and prognosis of complex systems, statistical signal processing and applications.

Received: 28 March 2023
Revised: 3 September 2023
Accepted: 2 October 2023