Condition monitoring of electric-cam mechanisms based on Model-of-Signals of the drive current higher-order differences

Matteo Barbieri * Roberto Diversi * Andrea Tilli *

* Department of Electrical, Electronic, and Information Engineering, University of Bologna, Viale del Risorgimento 2, 40136 Bologna, Italy (e-mail: [matteo.barbieri15, roberto.diversi, andrea.tilli]@unibo.it).

Abstract: Condition monitoring of electric motor driven mechanisms is of great importance in industrial machines. The knowledge of the actual health state of such components permits to address maintenance policies which results in better exploitation of their actual operational life span and consequently in maintenance cost reduction. In this paper, we exploit the way electric cams are implemented on the vast majority of PLC/Motion controllers to develop a suitable condition monitoring procedure. This technique relies on computing the higher-order differences of the current absorbed by slave motors to get signals that do not depend on a priori knowledge of the cam trajectory and of the mechanism nominal model. Subsequently, we will use these data in the Model-of-Signals framework, to gather information on the mechanism's health condition, which in turn can be used to perform predictive maintenance policies. The differenced signal is modelled as an ARMA process and the model capabilities in condition monitoring are then shown in simulation and experimental application. Besides, this framework allows exploiting the edge-computing capabilities of the machinery controllers by implementing recursive estimation algorithms.

Keywords: Condition Monitoring, Electric Drives, Programmable Logic Controllers, Edge-Computing, Fault diagnosis, Predictive Maintenance, Prognosis and Health Management, Industry 4.0.

1. INTRODUCTION

Prognostic and Health Management (PHM) of systems within the Industry 4.0 framework has grown in importance so to become one of the main driving concepts in industrial automation firm R&D branches and in academia. Machinery parts are likely to fail when put through heavy-duty working cycles. Diagnosis of their health state and prognosis of their remaining useful life (RUL) is a widely addressed problem in the research field (Gouriveau et al., 2016). This has been made possible by the increased computational capacity of computers, both on-board (edge-computing) and outside the machine (PCs and cloud-computing).

Condition Monitoring (CM) refers to the evaluation of equipment or component health condition when the machinery is executing operations and jobs. Research is particularly oriented to the diagnosis and prognosis of faults occurring on machine main elements: from bearings to gears to drives' mechanical parts to electrical equipment. We refer to (Lee et al., 2014) for a comprehensive discussion, however, it is possible to summarise the main methods employed in condition monitoring within three main typologies depending on how they exploit (mathematically) the physical knowledge related to systems: Model-Based, Data-Driven and Hybrid methods.

Model-Based methods (Isermann, 2005) rely on physical modelling to build mathematical approximations, of increasing degree of complexity, to characterise its input/output behaviour. These methods run in parallel with the machinery under monitoring to provide information on the internal state of the machine. Physical modelling is particularly effective in terms of diagnosis and prognosis of faults, however its complexity may

result prohibitive for edge-computing, particularly concerning the automatic machine field. Data-Driven methods (Cerrada et al., 2018) exploit signals measured on-board to perform CM, mainly by means of signal processing and machine learning techniques. The implementation of such strategies is simpler and requires, in general, less time and resources. Finally, hybrid methods combine the previously mentioned ones.

Any monitoring procedure requires significant sensor measurements, suitable data processing algorithms and appropriate servicing choices (either automated or with human intervention). Condition-Based Maintenance (CBM) (Jardine et al., 2006) and Predictive Maintenance (PM) encapsulate formally those concepts by defining a broader picture of the course of actions involved by dividing it into three macro-steps

(1) Data acquisition.
(2) Data processing.
(3) Maintenance decision-making.

In this work, we focus mostly on the first two steps supplying the foundations to perform the third. The technique we exploit belongs to the data-driven methods and employs black box system identification theory: it is referred to as Model-of-Signals and was introduced by (Isermann, 2006). As the name suggests, it relies on the signals measured on-board the monitored machine to build dynamic models by means of system identification algorithms (Söderström and Stoica, 1989). The main reasons behind the use of this approach revolve around two of its main properties: the models carry inherent information about the system physical content and the availability of recursive algorithms permits the implementation directly on the PLC, exploiting its edge-computing capabilities. Furthermore,

Model-of-Signals methods compress signals information into models that are easier to handle. This allows the use of distributed computing frameworks with models becoming features for other data-driven algorithms.

Following our previous studies on Model-of-Signals (Barbieri et al., 2018, 2019a,b), we present a different application of the approach that aims at exploiting how electric cams are executed on PLCs to monitor electric motor driven mechanism conditions employing applied torque as source of information. The use of the torque (i.e current) measurement for monitoring electric motor components (e.g. cage, windings, resistance, bearing and shaft faults) is long-established (Nandi et al., 2005) and typically makes use of model-based, frequency-based and data-driven techniques with high-end servo drives starting to include them within their controllers. However, due to the unknown task they will have to accomplish in their final application, the above mentioned techniques are only related to the motor internal health state and not to the mechanism attached to it, which is also subject to failure. The condition monitoring technique we propose aims at providing the manufacturer with a PLC practicable solution for drive-mechanism fault detection.

The main idea is the following: the majority of electric cam motion tasks for servo drives are implemented as piece-wise polynomial trajectories with order lower or equal to 7. Therefore, in the case of linear mechanisms the ideal torque demanded by their motion is linked to the second derivative of such curves. In real applications, however, another component is present alongside the ideal torque: smaller with respect to the latter, but necessary to achieve the desired motion. Our conjecture on that additional contribution is that it contains information about the mechanism health condition and it can be modelled by a set of Auto-Regressive (AR) models. Its analysis require the ideal contribution to be removed in order to prevent it from masking changes within the useful one (in this domain, the ideal torque is the "noise" perturbing the informative signal). A simple subtraction of the ideal torque could be arranged in this respect, however, it would depend on the given cam trajectory and on the equivalent inertia of the mechanism. In this work, we propose to compute the difference of suitable order of the torque measurement (which is linked to the order of the trajectory polynomial profile minus 2, therefore 5 at worst) to get rid of its ideal contribution without any cam and mechanism detailed knowledge. Then, the useful part of the signal will be modelled as an AR process, which will be proven to became an Auto-Regressive Moving-Average (ARMA) with particular structure when differencing. This allows Model-of-Signals to be applied following its basic idea as in Barbieri et al. (2018, 2019a,b).

The remainder of the paper is organised as follows: Section 2 describes the reasoning behind our condition monitoring proposition using slave torque. Then, in Section 3, the signal modelling and identification approach is illustrated and tested via simulation in Section 4. In Section 5 we apply the proposed procedure to real data in a laboratory setup whose outcomes are shown in Section 6. Finally, conclusions are drawn in Section 7.

2. FROM ELECTRIC CAMS TO DIAGNOSIS

The majority of industrial machines rely on cams to perform complex tasks that require synchronisation among the various mechanisms involved. Cams can be divided into mechanical and electrical. The use of the latter to perform synchronised op-
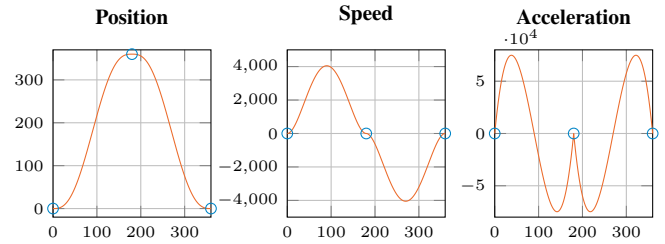

Fig. 1. Example of piece-wise polynomial cam: via-points in blue, master and slave positions are in degrees $[deg]$, while the acceleration is in $[deg^{-1}]$

erations is increasing in the last decades due to their comparable precision and greater flexibility, with respect to mechanical ones. Electric cams allow to coordinate the motion of different mechanisms independently driven by electrical motors. This is possible because servo drives have become able to precisely track given position profiles commanded via fieldbus by the PLC, allowing the synchronisation of movements via software.

2.1 Electric cams PLC implementation

Electric cams are performed by linking together the trajectories of the different motors involved in the synchronised task: a leader, known as master, performs the guiding trajectory while one or more followers, called slaves, move accordingly. The coupling is established geometrically so that any given master trajectory point corresponds to a given slave trajectory point. This coupling is usually programmed by the user on the PLC vendor Integrated Development Environment (IDE). The typical implementations rely on the definition of via-points within the trajectory, which are then connected through mathematical functions which depend on the trajectory constraints. In most cases, polynomial functions, with their smoothness degree dependent on the number of constraints, are used. The constraints, in this case, originate from the required trajectory derivatives at those via-points. For instance, to build a master-slave synchronisation we need the master trajectory in position, $p(\cdot)$ and the relative slave position evolution, defined as $q(p(\cdot))$. This definition allows to geometrically connect the two trajectories, while time enters indirectly with the master position, allowing speed variations without affecting synchronisation. Obviously, also the physical limits of the system affect the trajectory (e.g. the maximum allowed speed and acceleration) and have to be taken into account during the design phase. An example of synchronisation definition procedure is given as follows:

$$q_1(0°) = 0°, \quad q_2(180°) = 360°, \quad q_3(360°) = 0°,$$
$$\dot{q}_1(0°) = 0, \quad \dot{q}_2(180°) = 0, \quad \dot{q}_3(360°) = 0,$$
$$\ddot{q}_1(0°) = 0^{°^{-1}}, \quad \ddot{q}_2(180°) = 0^{°^{-1}}, \quad \ddot{q}_3(360°) = 0^{°^{-1}},$$
(1)

in which q_1 is connected to q_2 with a polynomial function of order 5 since there is a total of 6 constraints. The same reasoning can be done for the cam piece between q_2 and q_3, with the final result shown in Fig.1. If we assume that master speed is constant (as typically happens in real applications), $\dot{p}(t) = const = V_p$, then the x-axis can be directly translated in time by means of $t = p(t)/V_p$. We refer to (Biagiotti and Melchiorri, 2008) for a complete discussion on how trajectories are generated.

2.2 The Torque for Monitoring

Suppose that the controller of the motors we want to synchronise is correctly designed and tuned. The master operates at

constant speed followed by the slave with a trajectory defined as in (1) driving a linear mechanism. The ideal torque required to perform the task in this case is:

$$\tau(t) = J\alpha(t) = J\ddot{q}(t), \qquad (2)$$

where $J > 0$ is the moment of inertia and $\alpha(t) = \ddot{q}(t)$. As a consequence, $\tau(t)$ is a piece-wise polynomial trajectory based on M couples of master-slave points and their constraints with the former converted into their time counterparts $t \in [t_1 \ \dots \ t_M]$ following the constant speed assumption. Therefore, ideal torque trajectory segments correspond to the second derivative of the related position profile piece scaled by the inertia factor J. This can be formally described as follows:

$$\tau(t) = \begin{cases} \mathcal{P}^1_{k(1)}(t) & t \in [t_1, t_2] \\ \vdots & \vdots \\ \mathcal{P}^m_{k(m)}(t) & t \in [t_m, t_m+1] , \\ \vdots & \vdots \\ \mathcal{P}^{M-1}_{k(M-1)}(t) & t \in [t_{M-1}, t_M] \end{cases} \qquad (3)$$

where $m = 1, \dots, M - 1$ is the index of the polynomial piece represented as $\mathcal{P}^m_{k(m)}(t)$ with degree $k(m) = d(m) - 2$, where $d(m)$ is the degree of the respective position polynomial.

The torque measurement from the slave axis is readily available in PLCs implementing electrical cams. Typically, this signal carries the ideal torque profile required by the mechanism, as in (2), with parametric uncertainties in J in addition to unmodelled ones (e.g friction, control adjustments and induced vibrations). As stated in the introduction, our conjecture is that information about the machine state of health is contained in this unknown part. If we are able to take out the ideal cam contribution, the remaining signal can then be used in the Model-of-Signals fashion to perform diagnosis. Our proposition starts from the idea that computing the $(k + 1)^{th}$ difference of the ideal torque profile (with k the maximum degree of the polynomials in $\tau(t)$), will result in a zero signal. Then, in the real case, we propose to model the torque measurement as an AR process containing information about the system plus the cam nominal torque, which we can get rid of by computing the $(k + 1)^{th}$ difference. Therefore we consider that AR process as representative of the machine health state. The difference computations affect the AR process turning it into an ARMA process. Nevertheless, it is still possible to extract that piece of information with recursive system identification algorithms, as we show in the next section. Notice that it may be advisable, in the difference calculation, to avoid the cam via-points, where the discontinuities generate steps and impulses. This filtering will be used in simulation, while in the real case it is not required since the physics of the system directly filters this contribution.

3. DEFINITION AND IDENTIFICATION OF THE SIGNAL MODEL

Following our reasoning we assume the torque signal $\tau(t)$ to be composed by polynomials with the addition of an AR process:

$$\tau(t) = P^m_{k(m)}(t) + e(t), \qquad (4)$$

with $P^m_{k(m)}(t)$ compactly denoting (3) and

$$e(t) = -a_1 e(t-1) - \dots - a_n e(t-n) + w(t) = \frac{w(t)}{A(z^{-1})}, \qquad (5)$$

is an AR process of order n with driving white noise $w(t)$ and $A(z^{-1}) = 1 + a_1 z^{-1} + \dots + a_n z^{-n}$, where z^{-1} is the

backward-shift operator, i.e. $z^{-1}e(t) = e(t-1)$. From now on we drop the symbol $k(m)$ to address the polynomial degree and just use k for the sake of clarity. This modelling permits to take into account the uncertainties derived from the real case. Now, if we apply the difference operator $(1 - z^{-1})$ to (2) we obtain

$$(1 - z^{-1})\, \tau(t) = (1 - z^{-1}) \left[P^m_k(t) + \frac{w(t)}{A(z^{-1})} \right] \qquad (6)$$

which becomes

$$\tau^{(1)}(t) = P^m_{k-1}(t) + \frac{(1 - z^{-1})w(t)}{A(z^{-1})}. \qquad (7)$$

Then, by applying again the difference operator k times, we get the $(k + 1)^{th}$-order difference of $\tau(t)$:

$$\tau^{(k+1)}(t) = \frac{(1 - z^{-1})^{k+1}w(t)}{A(z^{-1})}, \qquad (8)$$

in which the polynomial contribution disappears and the remaining part is nothing but the following ARMA model:

$$y(t) = \tau^{(k+1)}(t) = \frac{D(z^{-1})}{A(z^{-1})}w(t), \qquad (9)$$

where the coefficients of $D(z^{-1})$ are known and correspond to the ones of the binomial expansion of $(1 - z^{-1})^{k+1}$.

At this point, the problem to be solved consists in estimating the coefficients of $A(z^{-1})$ that, as already said, provide information about the health state of the machine. This identification problem can be solved by employing the Instrumental Variable (IV) method (Söderström and Stoica, 1989; Ljung, 1999). The adopted identification procedure can be summarised as follows. First, the signal $y(t)$ is obtained by performing the $(k + 1)^{th}$ difference of $\tau(t)$ in (8) and its expression in (9) becomes:

$$y(t) = -a_1 y(t-1) - \dots - a_n y(t-n) +$$
$$+ (-1)^0 \binom{k+1}{0} w(t) + (-1)^1 \binom{k+1}{1} w(t-1) + \dots$$
$$+ (-1)^{k+1} \binom{k+1}{k+1} w(t-k-1), \qquad (10)$$

which in regressor form is:

$$y(t) = \varphi^T_y(t)\theta_A + \varphi^T_w(t)\theta_D \qquad (11)$$

with

$$\varphi_y(t) = [-y(t-1) \ \dots \ -y(t-n)]^T \qquad (12)$$
$$\varphi_w(t) = [w(t) \ \dots \ w(t-k-1)]^T \qquad (13)$$
$$\theta_A = [a_1 \ \dots \ a_n]^T. \qquad (14)$$

Now, if we choose the following vector of instruments

$$\bar{\varphi}_y(t) = [-y(t-k-2) \ \dots \ -y(t-k-1-n-q)]^T, \quad (15)$$

with $q \geq 0$, the IV estimate of θ_A, when N samples of $y(t)$ are available, is given by

$$\hat{\theta}_A = \hat{R}^+ \hat{\rho}, \qquad (16)$$

where

$$\hat{R} = \sum_{\tau=\tau_0}^{N} \bar{\varphi}_y(\tau)\varphi^T_y(\tau), \quad \hat{\rho} = \sum_{\tau=\tau_0}^{N} \bar{\varphi}_y(\tau)y(\tau), \qquad (17)$$

$\tau_0 = n + q$, and $\hat{R}^+$ denotes the pseudoinverse of the matrix $\hat{R}$. The choice of the instrument vector (15) guarantees the consistency of the estimate for $N \rightarrow \infty$ (Söderström and Stoica, 1981, 1989). Note that if $q > 0$, the solution (16) becomes an extended IV method (Söderström and Stoica, 1989).

In this paper, we make use of the extended IV version proposed in (Friedlander, 1984), called Overdetermined Recursive Instrumental Variable (ORIV), which happens to be a robust recursive
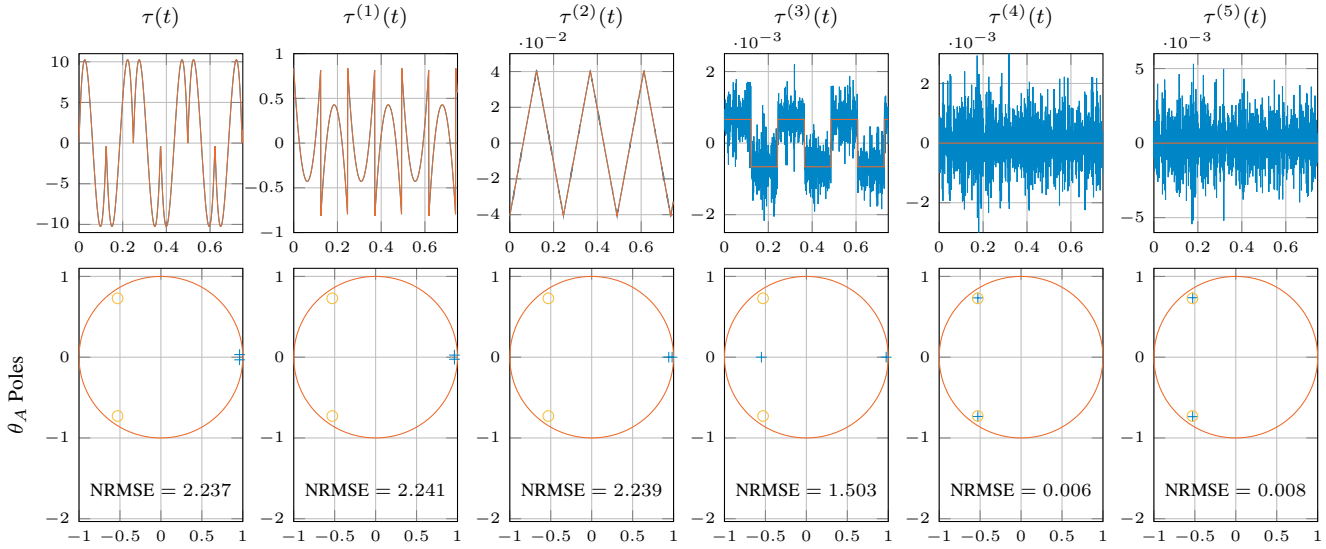
Fig. 2. [Top] Torque signal obtained as in (2) of 3 cams until the 5th difference. The ideal signal is in orange and the noisy one in blue. [Bottom] Poles of the identified $\hat{\theta}_A$ for the noisy signal (blue), of the applied noise (violet). Notice: the difference signals are processed so to avoid impulses to appear on the cam via-points.

form suitable for PLC implementation. In (Barbieri, 2017) it is shown how to realise the Recursive Least Squares (RLS) in machinery controllers, therefore some slight modifications are required to build the ORIV. To introduce the recursive version of the OIV algorithm we first rewrite (16) as follows:

$$\hat{\theta}_A(t) = \hat{P}(t)\hat{R}^T(t)\hat{\rho}(t) \tag{18}$$

with

$$\hat{\rho}(t) = \sum_{\tau=\tau_0}^{t} \bar{\varphi}_y(\tau)y(\tau) \tag{19}$$

$$\hat{R}(t) = \sum_{\tau=\tau_0}^{t} \bar{\varphi}_y(\tau)\varphi_y^T(\tau) \tag{20}$$

$$\hat{P}(t) = \left[\hat{R}^T(t)\hat{R}(t)\right]^{-1}. \tag{21}$$

Finally, the recursive version of the algorithm is the following.
Algorithm 1. (ORIV).

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

The initial step may be defined in the following way

$$\begin{matrix} \hat{\theta}_A(0) = 0 & P(0) = \psi I \\ \rho(0) = 0 & R(0) = 0 \end{matrix} \tag{22}$$

with ψ any large positive number.

Note that the algorithm requires no inversion of a variable dimension $n \times n$ matrix, while the 2×2 matrix inversion at step 2) can be easily implemented.

4. SIMULATION

The validation of the proposed Model-of-Signals approach in simulation is done utilising the cam from the example (1) repeated 100 times, with a master working at constant speed $V_p = 1440[°/s]$. We compute the torque $\tau(t)$ required to perform the task with a linear mechanism as in (2) with inertia $J = 0.0044[kg/m^2]$. The sampling time adopted in this simulation is $T_s = 0.001[s]$, which is commonly used on PLCs implementing electric cams in their programs. The ideal torque signal, in this case, will be composed of polynomials of the 3rd degree:

$$\tau(t) = P_3^m(t). \tag{23}$$

The torque signal simulating a real case, as in (4), will also have an AR process of order $n = 2$ in addition, whose parameter vector is the following:

$$\theta_A = [1.058 \quad 0.81]^T, \tag{24}$$

with driving noise variance $\sigma_w^2 = 10^{-8}$ and the following couple of complex conjugate poles:

$$\sigma\left(A(z^{-1})\right) = [0.9e^{i0.7\pi}, 0.9e^{-i0.7\pi}]. \tag{25}$$

Finally, the simulation has been performed by applying Algorithm 1, with the hyperparameter $q = 2$. The Normalised Root Mean Square Error ($NRMSE$) index:

$$NRMSE = \sqrt{\frac{\left\|\hat{\theta}_A - \theta_A\right\|}{\|\theta_A\|}}, \tag{26}$$

has been used to evaluate the performance in obtaining a $\hat{\theta}_A$ as close as to the value set in (24). The results of the simulation are shown in Fig. 2. It is possible to observe how in the $(k + 1 = 4)^{th}$-order difference the ideal torque contribution (orange) turns to zero, while the real torque signal (blue) becomes a zero mean ARMA process characterised by the polynomial $A(z^{-1})$ and then by the same coefficients θ_A of the AR process, see (9) and (11). The application of Algorithm 1 to $\tau^{(4)}(t)$ results in the identification of the AR process added to the torque with an error of 0.6% ($NRMSE = 0.006$). Therefore, we are able to isolate the model of the AR signal and discard the nominal torque piece of information. In this way, given an electric
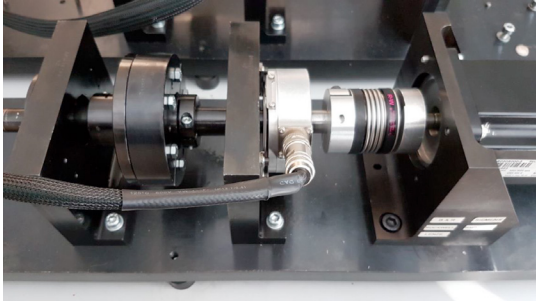
Fig. 3. Experimental linear rigid mechanism setup. Variable weights are attached to the flywheel on the left side of the shaft using bolts.

cam application, the contribution of the uncertainties may be extracted and used for condition monitoring in the Model-of-Signals framework.

5. EXPERIMENTAL SETUP

The experimental setup utilised is presented in Fig. 3 and is composed by, from right to left: electrical motor, rigid joint, shaft and flywheel with two half-moon shaped weights, the cabled encoder seen in the figure is not used in this experimental analysis and is not taken into account. The inertia of the system can be divided into two parts: one is fixed, with a value of $J_{fix} = 0.0015 [kg/m^2]$, and one is variable $J_w = J_{w_1} + J_{w_2}$ depending on the two weights, J_{w_1} and J_{w_2}, attached. The mechanism is driven by B&R equipment: the PLC is the `Automation PC 910` connected to an `ACOPOS P3` servo drive controlling a `8LSA36 DB030S000-3` brushless motor. The electric cam utilised is the same as in example (1) performed using a virtual master running at constant speed $V_p = 1080°/s$. The synchronised motion task time was chosen to be $T_s = 0.0008s$ since the system only allows time-steps of $0.0004s$ or multiples and that was the recommended setting. Therefore, the measurement of the torque signal has the same resolution and is collected by means of the tracing system provided by B&R IDE, Automation Studio, with a sampling frequency of $1250Hz$ and can be directly saved into `.mat` format. To test the proposed monitoring approach we sampled the slave drive torque during the synchronised motion of the system with both symmetric and asymmetric (i.e unbalanced) load. The former being the healthy reference operating point and the latter being the faulty one achieved by modifying one of the two half-moon shaped weights with a slightly thicker and a slightly thinner one. In addition, in the symmetrical case, one of the two weights has been loosened by sightly unfastening the bolts that keeps it in place to simulate a fault with increased degree of severity. Those unbalanced loads should generate changes in the informative part of the torque measurement which in turn should be captured by the models. The four tested configurations are depicted below:

Config. (1) Symmetric load:
$$J_{w_1} = 7.1305 \cdot 10^{-4}, \ J_{w_2} = 7.1305 \cdot 10^{-4} [kg/m^2] \quad (27)$$
Config. (2) Asymmetric increased load:
$$J_{w_1} = 7.1305 \cdot 10^{-4}, \ J_{w_2} = 7.5030 \cdot 10^{-4} [kg/m^2] \quad (28)$$
Config. (3) Asymmetric decreased load:
$$J_{w_1} = 7.1305 \cdot 10^{-4}, \ J_{w_2} = 6.1725 \cdot 10^{-4} [kg/m^2] \quad (29)$$
Config. (4) Loose Symmetric load: same as Config. (1) but with loosened bolts in one of the weights.

Various measurements of the slave torque were collected during operations in all configurations. Then, they were processed by

simulating a PLC implementation via Matlab. In particular, the main steps of the finite state machine involved in the processing are the following:

(1) Sample the slave torque in a buffer of $N = 10000$ elements;
(2) Compute the signal model with Algorithm 1;
(3) Compute the $NRMSE$ distance index, as in (26), with respect to the reference model.

The reference model is computed as the mean value of the first 10 models obtained during known healthy operating conditions by exploiting steps (1) and (2). Algorithm 1 is executed with the following hyperpameters: $n = 2$ as the AR model order, obtained with AIC criterion (Akaike, 1974), $q = 6$ more equations in its overdetermined part and $N = 10000$ as number of samples. Each model is the outcome of the algorithm after N sample are processed. The buffer and process architecture is applicable to any PLC since it allows the storage of N data samples to be coded within the main priority task, in this case with a sampling time of $Ts = 0.0008s$, and the implementation of their processing in a secondary task of lower priority, without affecting significantly the system memory and the control program computational load. This keeps the condition monitoring task on-line, still able to check machine health state with respect to degrading faults. For instance, a model is obtained every few seconds while mechanism degradation due to friction or wearing or heat typically takes minutes to hours to even days. In this fashion, the reference AR model computed while the system is in healthy working conditions is then compared with the models obtained while operating in the previously depicted load configurations.

6. RESULTS

The data collected from the depicted processing are here shown and analysed. The torque signal measured during healthy operations and its computed differences are shown in the top row of Fig.4. Due to the higher power of the AR part with respect to the ideal torque demand, the signals have approximately constant mean already at $\tau^{(2)}(t)$. This results in acceptable fault detection from the 2nd difference if we use the $NRMSE$ indicator with a threshold of $T_h = 0.1$, as we can see in the last row of Fig.4. However, in $\tau^{(2-3)}(t)$ case, a false healthy condition may appear for Config. (3), despite $\tau^{(2)}(t)$ being the best in pointing out Config. (4). This can be also deduced within the model plots, in the middle row of Fig.4, who show how close they are to the healthy ones. The best indicator in this respect is obtained in the case of $\tau^{(4)}(t)$ where faults have separate levels revealing satisfactory fault detection when T_h is applied. In particular, it turns out to be the most suitable to perform the monitoring task validating our proposition.

7. CONCLUSION

The aim of this work was to provide a procedure to monitor servomotor driven mechanisms in PLC controlled machines by means of their absorbed torque. We exploit the way electric cams are programmed to get rid of the nominal torque contribution by computing its $(k + 1)^{th}$-order difference and use the remaining signal as representation of the machine health state in the Model-of-Signal framework. This allows condition monitoring of the servo mechanism system without any detailed knowledge of the electric cam (only the maximum polynomial degree) and the related mechanism and equivalent inertia. Moreover, it does not require the addition of diagnosis sensors
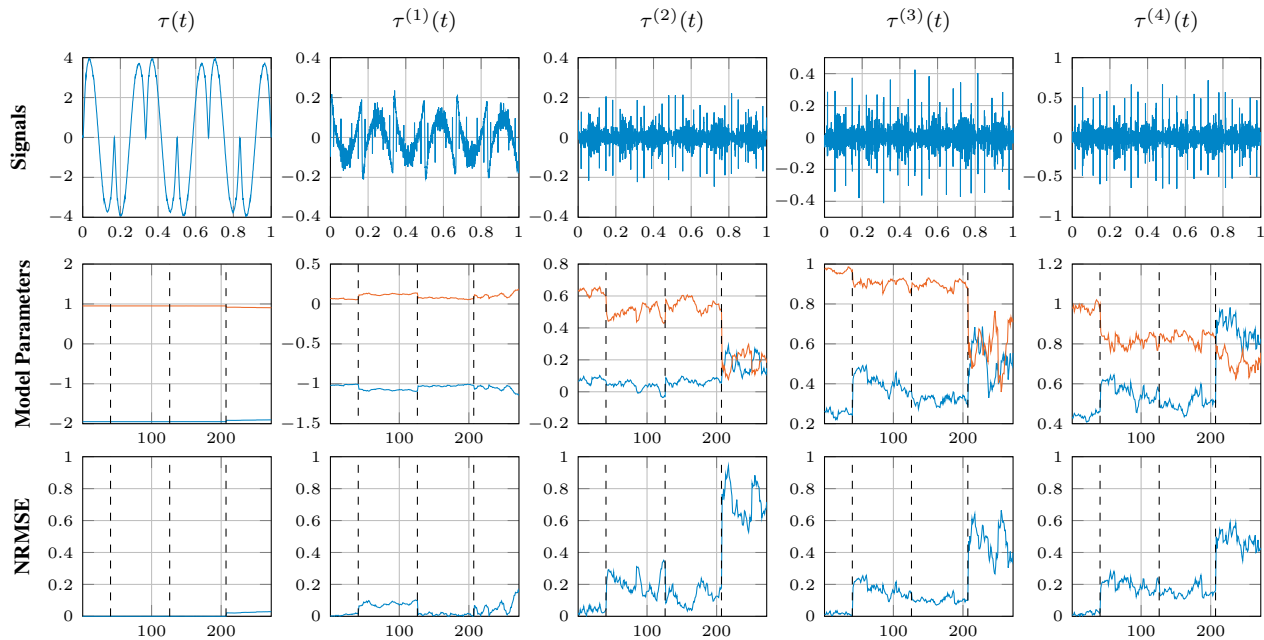
Fig. 4. [Top] Signals: the healthy signal of 3 cams and its computed differences are shown. [Middle] Models: the collected models for the four configurations are shown. [Bottom] NRMSE: the collected indexes are shown, a_1 in blue and a_2 in orange. The dashed lines divide one configuration from the other. They are, from left to right, Config. (1) to (4) respectively.

on board and permits to perform fault detection locally on the PLC by means of recursive algorithms. Finally, the models obtained can be fed, in a networked architecture, to computer able to draw more information and add fault isolation and prognosis providing the foundations to build intelligent maintenance systems. The next steps are involving the actual coding of the procedure and its employment in industrial prototypes.

ACKNOWLEDGEMENTS

REFERENCES

Akaike, H. (1974). A new look at the statistical model identification. In *Selected Papers of Hirotugu Akaike*, 215–222. Springer.

Barbieri, M. (2017). *Seamless infrastructure for "Big-Data" collection and transportation and distributed elaboration oriented to predictive maintenance of automatic machines.* Master's thesis, University Of Bologna.

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 *Enterprise Interoperability: Smart Services and Business Impact of Enterprise Interoperability*, 263–269. Wiley – ISTE.

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

Barbieri, M., Mambelli, F., Diversi, R., Tilli, A., and Sartini, M. (2019b). Condition monitoring by model-of-signals: Application to gearbox lubrication. In *15th European Workshop on Advanced Control and Diagnosis, ACD 2019.*

Biagiotti, L. and Melchiorri, C. (2008). *Trajectory planning for automatic machines and robots.* Springer Science & Business Media.

Cerrada, M., Sánchez, R.V., Li, C., Pacheco, F., Cabrera, D., de Oliveira, J.V., and Vásquez, R.E. (2018). A review on data-driven fault severity assessment in rolling bearings. *Mechanical Systems and Signal Processing*, 99, 169–196.

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

Gouriveau, R., Medjaher, K., and Zerhouni, N. (2016). *From prognostics and health systems management to predictive maintenance 1: Monitoring and prognostics.* John Wiley & Sons.

Isermann, R. (2005). Model-based fault-detection and diagnosis–status and applications. *Annual Reviews in control*, 29(1), 71–85.

Isermann, R. (2006). *Fault-diagnosis systems: an introduction from fault detection to fault tolerance.* Springer Science & Business Media.

Jardine, A.K., Lin, D., and Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical systems and signal processing*, 20(7), 1483–1510.

Lee, J., Wu, F., Zhao, W., Ghaffari, M., Liao, L., and Siegel, D. (2014). Prognostics and health management design for rotary machinery systems – Reviews, methodology and applications. *Mechanical systems and signal processing*, 42(1-2), 314–334.

Ljung, L. (1999). *System identification: theory for the user.* Prentice-hall.

Nandi, S., Toliyat, H.A., and Li, X. (2005). Condition monitoring and fault diagnosis of electrical motors - A review. *IEEE transactions on energy conversion*, 20(4), 719–729.

Söderström, T. and Stoica, P. (1981). Comparison of some instrumental variable methods – Consistency and accuracy aspects. *Automatica*, 17(1), 101–115.

Söderström, T. and Stoica, P. (1989). *System Identification.* Prentice Hall.