Condition Monitoring of a Paper Feeding Mechanism Using Model-of-Signals as Machine Learning Features

Matteo Barbieri[1], Francesco Mambelli[2], Jacopo Lucchi[2], Roberto Diversi[1], Andrea Tilli[1] and Matteo Sartini[2]

[1] *Department of Electrical, Electronic and Information Engineering "Guglielmo Marconi" (DEI), Bologna, Viale del Risorgimento, 2, 40136, Italy*
matteo.barbieri15@unibo.it
roberto.diversi@unibo.it
andrea.tilli@unibo.it

[2] *LIAM LAB, Vignola (MO), Via Confine, 2310, 41058, Italy*
francesco.mambelli@liamlab.it
jacopo.lucchi@liamlab.it
matteo.sartini@liamlab.it

Abstract

Prognostics and Health Management of machine devices and parts is a hot topic in the Industry 4.0 era. In this fashion, automated procedures to evaluate machinery working conditions are essential to minimize downtime and maintenance costs. In this work, we study how to monitor the decrease in performance of a paper sheet feeder for the packaging industry under heavy-duty cycle operations. The main measurable outcome of such degradation is the increase in backlash among the device moving components. A wide variety of methods and procedures is available to tackle this monitoring problem. In this paper, we analyze the use of a simple yet efficient diagnosis methodology that can exploit machinery controllers (i.e., Programmable Logic Controllers) edge-computing capabilities. Vibration measurements are known in the literature to retain information about the system's mechanics. Model-of Signals, a data-driven approach based on black box system identification, allows to extract that information reliably during machinery working cycle. The refinement of those data using machine learning allows the retrieval of knowledge about the health state of the machine. In this study, the feeder mechanism is run to failure with its parts backlash measured at given time intervals. Accelerometer signals are modelled as AutoRegressive processes whose coefficients are then considered as features to feed to machine learning algorithms, which are employed to perform severity evaluation of the ongoing degradation. Estimation and prediction are both implementable on-board the controller, while the learning task can be carried out remotely, in a cloud computing perspective. The exploitation of AutoRegressive modelling gives a simple and inherent methodology for feature selection, serving as a foundation of the machine learning stage. We make use of a Support Vector Machine algorithm to analyze how obtained models represent the various levels of backlash in the device and develop a suitable predictor of the degradation severity. Finally, the results of the application of the methodology to the case study are shown.

1. Introduction

Prognostics and Health Management (PHM) of machines, in recent years, has become a determinant factor in the industrial world, especially for firms adopting the main concepts of Smart Factory and Intelligent Manufacturing. In this context, autonomous diagnostics and prognostics of faults and their precursors has gained remarkable attention. The field is flourishing in academia, and researchers have published numerous PHM methodologies for machinery components (Lee et al., 2014; Atamuradov, Medjaher, Dersin, Lamoureux, & Zerhouni, 2017; Vogl, Weiss, & Helu, 2019).

Autonomous health management is based on Condition Monitoring (CM), which refers to tracking the equipment state of health during operations. On top of that, it is possible to build-up maintenance policies, such as Condition-Based Maintenance (CMB) (Jardine, Lin, & Banjevic, 2006) and Predictive Maintenance (PM) (Javed, Gouriveau, & Zerhouni, 2017). The former is usually triggered when a monitored device reaches a certain level of degradation, while the latter depends on the component predicted level of deterioration in time. The typical course of actions adopted to execute those servicing strategies involves:

1. Sensor data acquisition.
2. Data processing.
3. Maintenance decision making.

They denote the fact that the application of autonomous health management procedures on machinery components requires significant sensor measurements, suitable data processing algorithms, and appropriate servicing choices. However, the majority of the proposed PHM solutions do not take into account the standard hardware and software used in the industry. Those procedures typically rely on non-industrial equipment and software to produce useful information for servicing.

On the other hand, the industrial world is starting to integrate a large number of Information Technology solutions to keep up with the previously depicted concepts, starting from hardware. Modern machinery controllers are built based on PCs and workstations hardware architectures, introducing more computational power and resources in production lines. Moreover, they now offer enhanced connectivity, allowing improved interconnections within the automation pyramid. This development enables the use of Industrial PCs as edge-computing units, hence, their integration in autonomous health management procedures. Firstly, real-time sensor data collection through fieldbus is boosted to higher sampling frequencies, up to 10 kHz, fulfilling the requirements of the acquisition step. Secondly, additional CPU power enables local data processing, providing the first refinement stage of the collected information. Lastly, controllers can exploit increased networking capacity to outsource the final elaborations to production supervisor PCs, offering useful information for maintenance decision-making.

In this scenario, it is crucial to choose a reasonable data processing procedure to perform condition monitoring of the machinery components exploiting industrial PCs. In general, PHM proposes methods that involve model-based (Isermann, 2005) and data-driven (Cerrada et al., 2018) solutions. Among those, machinery controllers can host feasibly simple model-based solutions as well as simple data-driven ones. In this respect, we have chosen Model-of-Signals (MoS) (Isermann, 2006): A hybrid technique that makes use of black box system identification (Söderström & Stoica, 1989; Ljung, 1999), a well-established theory, to derive models from the system measured signals. As shown in (Barbieri, 2017; Barbieri et al., 2018), recursive identification algorithms implementation suits Industrial PCs programming languages, even though they are not meant for complex math operations. Such a technique allows us to compress sensor data streams into models, retaining most of the original information content. Then, the increased connectivity with higher-level computers permits the integration and processing of this information through data-driven approaches, such as Machine Learning (ML). The result is a two-level architecture producing PHM indications in a distributed fashion.

This work is a follow-through of previous projects (Barbieri, Diversi, & Tilli, 2019; Barbieri, Mambelli, Diversi, Tilli, & Sartini, 2019). In this one we study the application of the proposed two-level architecture to monitor the working condition of a paper feeder mechanism using MoS and Support Vector Machines (SVM). The system feeding quality is a consequence of the increase in backlash between its parts. This quantity degrades over time throughout working machine cycles. During production, accelerometers measure system vibrations, and their related models' generation is performed using MoS. Vibrations can be represented as an AutoRegressive (AR) process and the Recursive Least Squares (RLS) algorithm can be used to estimate its parameters. Meanwhile, mechanism backlash has been evaluated and recorded periodically during production stops. Then, the collected models have been labeled using the measured degradation level and fed to an SVM classifier. In other projects (Barbieri, Diversi, & Tilli, 2020), we use different underlying MoS structure, depending on the source of the signal to model. In that particular case, we model the current readings from the motor driving an electric cam mechanism as an AutoRegressive MovingAverage (ARMA) process.

In the remainder of the paper, section 2 is devoted to the presentation of our condition monitoring methodology and its mathematical foundations. Then, section 3 describes its application to the industrial case study with section 4 presenting the obtained results. Finally, PHM considerations are drawn on top of obtained results in section 5.

2. CONDITION MONITORING IN MACHINERY

The condition monitoring technique we use in this work exploits the standard, founding components of the automation pyramid, allowing manufactures to employ expertise they already own to implement the required elaborations. In PC-supervised production lines, the controllers have room for the implementation of system identification algorithms. In their suited form, they reduce the data load on the next computing level by compressing measurements streams into compact pieces of information, i.e., model parameters. Then, Programmable Logic Controllers (PLCs) boosted connectivity permits to send this polished data to supervising calculators, outsourcing the final refinements through the use of SVM, in this case, and finally, displaying the monitoring results for maintenance decision-making. In this connected scenario, we incorporate also a model distance metric in our PHM solution. Despite minor diagnostics capabilities with respect to machine learning algorithms, it has the advantage of providing a locally computed indicator for fail safe policies. The combination of this generated information is useful for the management level of the automation pyramid to drive a possible optimisation of the machinery maintenance strategies.

2.1. Model-of-Signals

The Model-of-Signals approach consists in assuming a mathematical model of the measured signal and estimating it from the available data. This estimate is then used to extract information for fault diagnosis (Isermann, 2006). In this work, the procedure relies upon signals sampled from accelerometers placed on-board the machine. The measured signals are assumed to be described by stochastic AutoRegressive processes that are estimated by a system identification algorithm, with the assumption that the machinery production process generates the AR "driving" noise.

In detail, the signal $y(t)$ coming from a sensor is assumed to be generated by an AR model described by the equation

$$y(t) = -a_1\, y(t-1) - \ldots - a_n\, y(t-n) + e(t), \tag{1}$$

where n is the model order and $e(t)$ is the "driving" white process. This model can be rewritten in the regression form

$$y(t) = \varphi^T(t)\,\theta + e(t), \tag{2}$$

where

$$\varphi(t) \;=\; \left[\, -y(t-1) \,\cdots\, -y(t-n) \,\right]^T, \tag{3}$$

$$\theta \;=\; \left[\, a_1\, a_2\, \cdots\, a_n \,\right]^T. \tag{4}$$

The identification problem consists in estimating the parameter vector θ given a set of samples of $y(t)$. This problem can be solved by means of the Least Squares (LS) algorithm (Ljung, 1999). However, the batch LS method is not suitable for PLC implementation since real-time controllers have both memory and computational constraints. For this reason, the Recursive Least Squares algorithm has been adopted, as described in the next subsection.

2.2. Identification algorithm

The Recursive version of the LS algorithm that makes use of the matrix inversion lemma (Ljung, 1999; Söderström & Stoica, 1989) results in being the best choice. In fact, it is possible to overcome computational constraints and part of the memory constraints of the conventional LS formulation, since the algorithm computations rely upon square matrices whose maximum dimension is the model order n.

The RLS algorithm computes the model parameter estimate $\hat{\theta}$ recursively in time. More precisely, the estimate $\hat{\theta}(t)$ at time t, is obtained on the basis of the previous estimate $\hat{\theta}(t-1)$, the current measurement sample $y(t)$ and the last n sensor measurements $y(t-1), \ldots, y(t-n)$. The algorithm is based on the following equations (Ljung, 1999):

$$\hat{\theta}(t) = \hat{\theta}(t-1) + K(t)\epsilon(t), \tag{5}$$

$$K(t) = \frac{1}{t} P(t)\,\varphi(t), \tag{6}$$

$$\epsilon(t) = y(t) - \varphi^T(t)\,\hat{\theta}(t-1), \tag{7}$$

$$P(t) = \frac{t\,P(t-1)}{t-1}\left[I_n - \frac{\varphi(t)\varphi^T(t)P(t-1)}{t-1+\varphi^T(t)P(t-1)\varphi(t)} \right], \tag{8}$$

where I_n is the $n \times n$ identity matrix. Then, notice that RLS has to be initialized by the quantities $\hat{\theta}(t_0)$ and $P(t_0)$ which have to be either defined by on-board calculations or sent to the machine. Those quantities are commonly computed in the following way, by performing a "mini-batch" LS,

$$P(t_0) = \Big(\frac{1}{t_0} \sum_{k=1}^{t_0} \varphi(k)\varphi^T(k) \Big)^{-1}, \tag{9}$$

$$\hat{\theta}(t_0) = P(t_0) \Big(\frac{1}{t_0} \sum_{k=1}^{t_0} \varphi(k) y(k) \Big), \tag{10}$$

with $0 \leq t_0 < t$. When not possible, e.g., in the majority of PLCs, they are initialised as $P(t_0) = \alpha I_n$ and $\hat{\theta}(t_0) = \mathbf{1}$, where $\alpha > 0$ is a scalar and $\mathbf{1}$ is a $n \times 1$ vector whose entries are all equal to 1.

The derived algorithm formulation is now suitable for PLC implementation and its on-line use during operations. As showed in (Barbieri, 2017), the algorithm is numerically validated showing the same performances when evaluated within MATLAB environment as well as on the machine controller. In our proposition, the algorithm and its initialization are both running on PLC programs. In this sense, depending on the PLC computational power and its available features, it is advisable to perform the condition monitoring program on a low priority task with suitable cycle time in order to let it work properly without affecting the main programs controlling the machinery operations. For instance, in the case study we are going to present, the MoS solution results optimal because of its light computational load since the involved logic control task is close to a 70% utilization factor.

2.3. Model Order Selection

The use of AR processes to represent the measured signals requires to define a proper model order n, see equation (1). Two criteria that are often used for model order selection are the Final Prediction Error (FPE) and the Minimum Description Length (MDL). They are based on the statistical properties of the residual of the LS identification. Consider an AR model of order n and the associated parameter vector $\hat{\theta}_n$ identified by applying the RLS method to a set of N measurements $y(1), y(2), \ldots, y(N)$. FPE and MDL are criteria with complexity terms that consist in selecting the order n leading to the minimum of the following loss functions (Söderström & Stoica, 1989; Ljung, 1999):

$$FPE(n) = \frac{N+n}{N-n} J(\hat{\theta}_n), \tag{11}$$

$$MDL(n) = N \log(J(\hat{\theta}_n)) + n \log N, \tag{12}$$

where

$$J(\hat{\theta}_n) = \frac{1}{N} \sum_{i=1}^{N} \varepsilon^2(t, \hat{\theta}_n), \tag{13}$$

and $\varepsilon(t, \hat{\theta}_n) = y(t) - \varphi^T(t)\, \hat{\theta}_n$ is the residual (prediction error) of the LS identification. In this work, the choice of the order n is performed by combining the FPE and MDL criteria with the whiteness hypothesis test applied to the residual sequence $\varepsilon(1, \hat{\theta}_n), \ldots, \varepsilon(N, \hat{\theta}_n)$. The whiteness test, which is often used for model validation (Söderström & Stoica, 1989; Ljung, 1999), is based on the following variable:

$$\xi_{N,m} = N \frac{\hat{r}_\varepsilon^{m^T} \hat{r}_\varepsilon^m}{J(\hat{\theta}_n)^2}, \tag{14}$$

where

$$\hat{r}_\varepsilon^m = \frac{1}{N} \sum_{i=1}^{N} \begin{bmatrix} \varepsilon(t-1, \hat{\theta}_n) \\ \vdots \\ \varepsilon(t-m, \hat{\theta}_n) \end{bmatrix} \varepsilon(t, \hat{\theta}_n). \tag{15}$$

More precisely, it possible to prove that $\xi_{N,M}$ is asymptotically chi-square distributed with m degrees of freedom:

$$\xi_{N,m} \xrightarrow[N \to \infty]{\text{dist}} \chi^2(m). \tag{16}$$

2.4. Distance Metrics

In this paper, we make use of the symmetric Itakura-Saito (IS) spectral distance (Wei & Gibson, 2000) to introduce a local metric for fault detection. The IS distance is a measure of how close to each other are the spectra of the estimated model and of the reference one. Since it compresses models information into a scalar it is not as reliable as SVM for fault severity isolation, but it is relevant for the definition of fail-safe policies within the machinery controller. The scenarios it allows to tackle, typically, are the loss of connection with the supervisor and unexpected degradation in the system. Its formulation is the following:

$$IS = \frac{1}{2\,N_f} \sum_{k=1}^{N_f} \left(\frac{S_{ref}(f_k)}{\hat{S}(f_k)} - \log \frac{S_{ref}(f_k)}{\hat{S}(f_k)} + \right. \\ \left. + \frac{\hat{S}(f_k)}{S_{ref}(f_k)} - \log \frac{\hat{S}(f_k)}{S_{ref}(f_k)} - 2 \right), \tag{17}$$

where $S_{ref}(f_k)$ denotes the Power Spectral Density (PSD) of the reference AR model θ_{ref}, $\hat{S}(f_k)$ is the PSD of the current estimated AR model $\hat{\theta}$ and N_f is the number of considered frequencies, i.e., the PSD resolution.

2.5. Support Vector Machines

Modelling signals as AR processes leads to a considerable reduction of the acquired data size, while the most relevant spectral content is captured. Therefore it is possible to use AR techniques as a feature selection method for machine learning algorithms, reducing both training and inference time and improving prediction accuracy. Given the collected dataset in the case study, we train a linear SVM (Vapnik, 1995; Cortes & Vapnik, 1995; Bishop, 2006) on the obtained models to predict the relative degradation level. The choice of this technique depends on how we were able to link the levels to the run to failure models in the dataset of the proposed case study. The evaluation of the play in the mechanism is a time-consuming process that only the machinery technician was allowed to perform, resulting in very few measurements with respect to the extracted AR features. In this fashion, we used as ML algorithm a classifier, and not a regressor to perform predictions.

Given a set of observations $\{x_i\}_{i=1,\ldots,N}$, $x_i \in \mathbb{R}^n$, and the relative labels $\{y_i\}_{i=1,\ldots,N}$, with $y_i \in \{-1, 1\}$ without loss of generality, we consider separation hyperplanes of the form

$$f(x) = w^T \cdot x + b = 0, \tag{18}$$

where the weights $w \in \mathbb{R}^n$ and the bias $b \in \mathbb{R}$ are the hyperplane parameters. A linear SVM is a binary classifier that aims to find the hyperplane that maximises the margin between the two classes. Hence, its goal is to determine the hyperplane parameters that maximises the distance from the support vectors of the two classes, i.e., from the observations nearest to the separation hyperplane. It results that

$$w = \sum_{x_i \in SV} \lambda_i y_i x_i, \tag{19}$$

$$b = \frac{1}{|M|} \sum_{x_i \in M} \left(y_i - \sum_{x_j \in SV} \lambda_j y_j x_i^T x_j \right), \tag{20}$$

where $\lambda = \left(\lambda_1 \ldots \lambda_N\right) \in \mathbb{R}_{\geq 0}^N$ are Lagrangian multipliers, SV is the set of support vectors and M is the set of support vectors whose corresponding λ_i are lower than the regularization parameter that penalises classification errors in case of non-separable classes.

In the proposed case study we have to deal with a classification problem with $k > 2$ classes, thus we adopt the one-versus-one approach, which is known to be robust with respect to this learning task (Bishop, 2006). It consists in training $\frac{K(K-1)}{2}$ binary SVMs on all the possible pair of classes and classifying the observations as belonging to the class that presents the higher number of assignments.

3. INDUSTRIAL CASE STUDY

The machinery under test in this study is an industrial paper feeder. It is a working group within a production line that involves paper sheet insertion in several packaging typologies. The device is subject to heavy-duty cycle operations, up to 30 000 cycles per hour and we kept this maximum value constant during our testing. One electric motor drives the whole mechanism. A system of gears, cams, belts and pulleys transmits the wanted synchronized motion to the end effectors, i.e., a combination of pliers and suction caps extracting paper sheets from a vertical stack. Even though the feeder is designed to work in high-performance conditions, its parts suffer from wearing over time. It causes the increase in play between its moving elements resulting in paper feeding quality degradation. We monitor such production deterioration by measuring device frame vibrations using two accelerometers, with an Industrial PC handling their acquisition and processing. Accelerometers are installed on the crank of the slider-crank mechanism that drives the pliers, oriented along the connecting rod, and on the shaft that releases the single paper sheets, oriented in the same direction of the motion of the vertical stack support, respectively. We cannot provide pictures of the mechanism because of the confidential nature of the project. Nevertheless, the firm we collaborate with has allowed us to share the obtained condition monitoring results.

3.1. Data Acquisition

The equipment used in this project reflects the considerations we introduced about the use of industrial PCs as edge-computing units. It consists of:

- C6920-0030: Beckhoff Industrial PC, with CPU Core2 Duo 2.53 GHz and RAM of 1 GB.
- Two PCB 353B03, i.e., monoaxial, piezoeletric, 500 g accelerometers with a measuring bandwidth of 1-7000 Hz and output signal of ± 5 V, that are acquired with one EL3632 module, which supports a maximum sampling frequency of 50 kHz. The accelerometers are labeled Acc1 and Acc2.

Components connection scheme is depicted in Figure 1. The PLC is responsible for the collection and the conditioning of sensor measurements. Controller and fieldbus main cycle times are synchronized and set to 1 ms, while the module sampling frequencies are configured to oversample at 5 kHz. Hence, at each cycle time, the controller receives 5 samples from each acquired sensor. The acquisition program groups accelerometer data into arrays with 18 420 entries resulting in 30 operating cycles.

```
C6920-0030:  Beckhoff Industrial PC
└─EK1100:  EtherCAT coupler
   └─EL3632:  IEPE analog input module
      ├─353B03:  Piezoelectric
      │  accelerometer 1
      └─353B03:  Piezoelectric
         accelerometer 2
```

Figure 1. Components connection scheme.

Cycles	Backlash [mm]
1 268 642	0.650
2 081 215	0.700
2 336 280	0.925
3 044 592	1.100
3 754 981	1.200
4 463 972	1.375
5 159 990	2.075
5 940 476	2.200
7 064 995	2.625
8 264 164	3.150
9 213 769	3.500
10 153 300	3.900

Table 1. Pliers backlash measurements with corresponding number of machine cycles reached at time of recording.

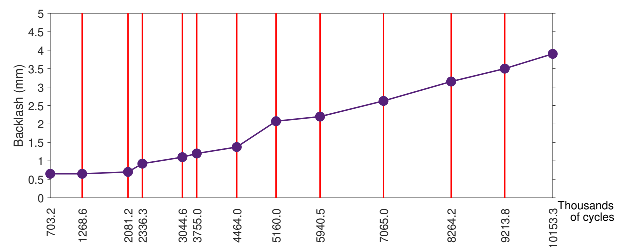

Figure 2. Plot of pliers backlash against the number of machine cycles reached at time of recording. The red lines indicate the time in cycles at which the measurement was taken.

We carried out this study by performing a run-to-failure test of the machine, stopping operations at given time instants to record the backlash level between parts, as shown in Table 1 and Figure 2. Despite the higher wearing registered between 4 and 5 million cycles, the degradation seems to increase constantly as time passes. The life span of such device is typically rated for about 10 000 000 total cycles, according to the vendor. In our testing, according to the feeder technician, we reached the highest possible level of play before the performance degradation was unsustainable, feeding a random number of sheets of paper and not one at a time. In practice, the definition of the non-return level of play depends on the customer and, in particular, on the machine mounted downstream the feeding group and the feeding quality and precision it requires.

3.2. Data processing

The data processing stage makes use of the PLC as an edge-computing unit and its supervisor as a remote-computing one.

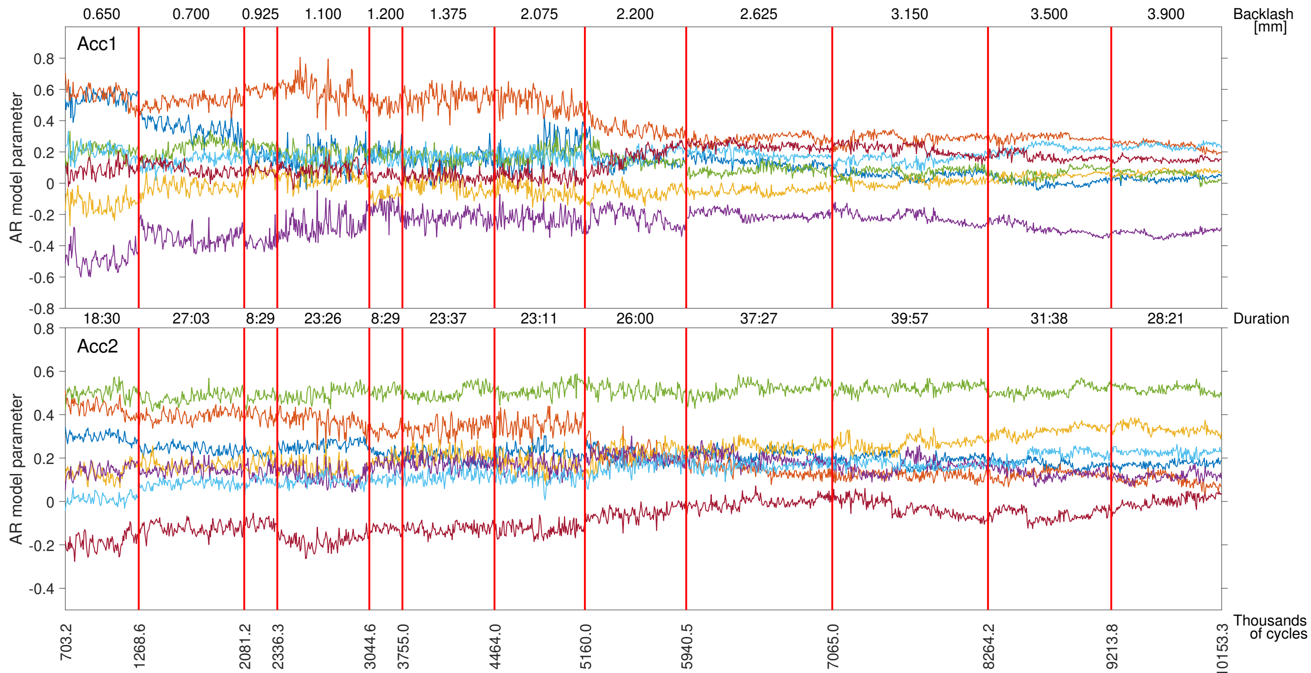

Figure 3. Model parameters evolution in time. Only the first 7 parameters of $\hat{\theta}$ are shown. Red lines separate the different working phases. The top, middle and bottom horizontal axes display the backlash level, the working phase duration and number of machine cycles reached, respectively.

The controller is responsible for the first information refinement, using the RLS algorithm to estimate the models of the buffered signals. The main hyperparameter to set is the model order. In this application, $n = 20$ is the result of the application of MDL and FPE on a sample signal measured during nominal working conditions with PLC available resources taken into account. Alongside this definition, the controller estimates the IS distance reference model. While performing the test, the industrial PC logs the estimated Acc1 and Acc2 models, together with the corresponding IS distance from the respective references and sends those value to the supervising computer using the MQTT communication protocol. Besides, the current backlash measure is recorded, with the operator doing this operation via Human-Machine Interface (HMI) every time the machine is stopped.

Once the test and data collection is over, the supervising PC, running MATLAB in this case, performs the machine learning task, providing the results that we will analyze in the next section. The adopted SVM algorithm exploits a linear kernel and penalises classification errors with a regularization parameter of 1. SVM labels data according to the recorded backlashes, which are however available only in correspondence of machine stops. Therefore, when the operator inserts a backlash value in the HMI, it labels all the data processed between the previous and the actual machine stop. It results in a 12-classes data partition. The algorithm partitions data with the same proportions within each class, by picking randomly 70 per cent of data for the training phase, 0 per cent for validation and 30 per cent for the test stage. Due to the high cost of such tests, the firm allowed us to record only one run to failure of the machine, so we had to train and test using this unique run only.

4. MONITORING RESULTS

Here we present the results obtained by using the methodology proposed in section 2 and applied in section 3. Initially, we analyze the data collected and processed on-board the PLC. Figure 3 presents the first 7 parameters of the signal models computed for both Acc1 and Acc2 during the run-to-failure test. Top, middle and bottom horizontal axes show the backlash level, the time duration and the number of cycles reached for each working period, respectively. Even if for the sake of clarity and space we do not show all of them, the graphs allow already a qualitative recognition of the different degradation stages of the mechanism. This suggests that the use of models as features for the automatic generation of PHM indicators is practicable for this condition monitoring solution.

To this extent, we provide also the data relative to IS distance measurement obtained throughout the testing in Figure 4. It shows that a scalar quantity, computed locally, can provide insight into the system state of health. It has an increasing

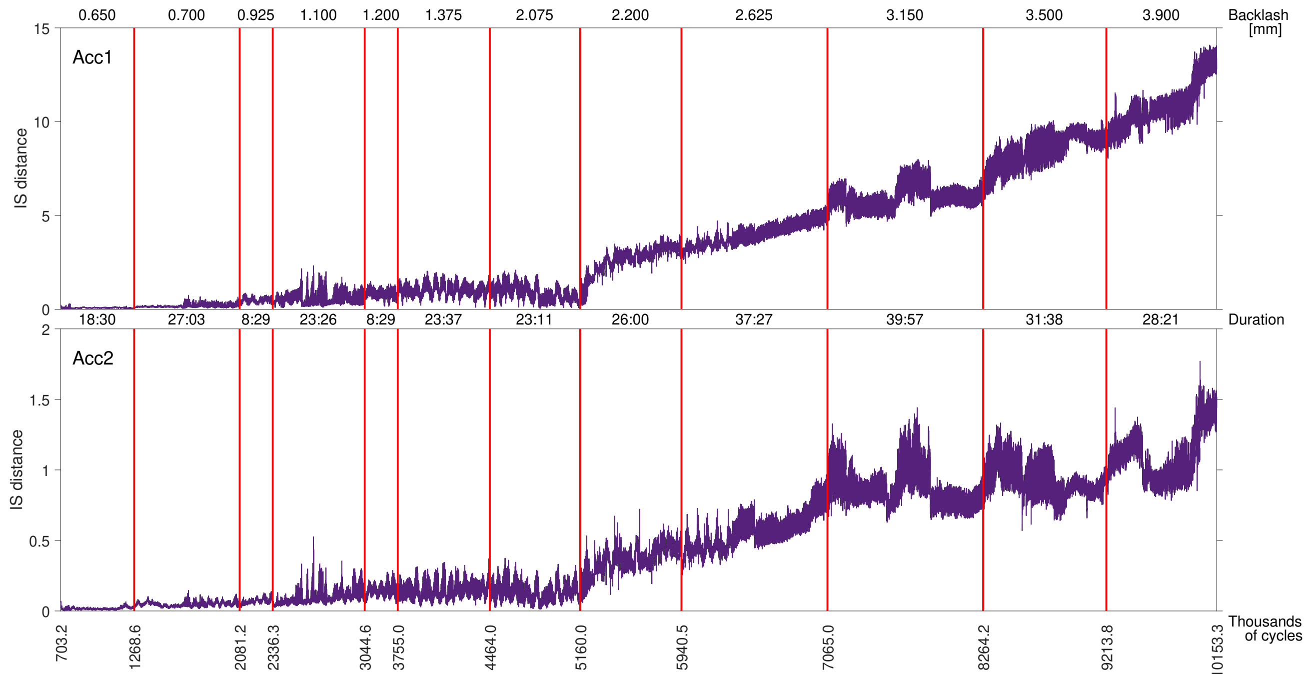

Figure 4. Itakura-Saito distance evolution in time. Red lines separate the different working phases. The top, middle and bottom horizontal axes display the backlash level, the working phase duration and number of machine cycles reached, respectively.

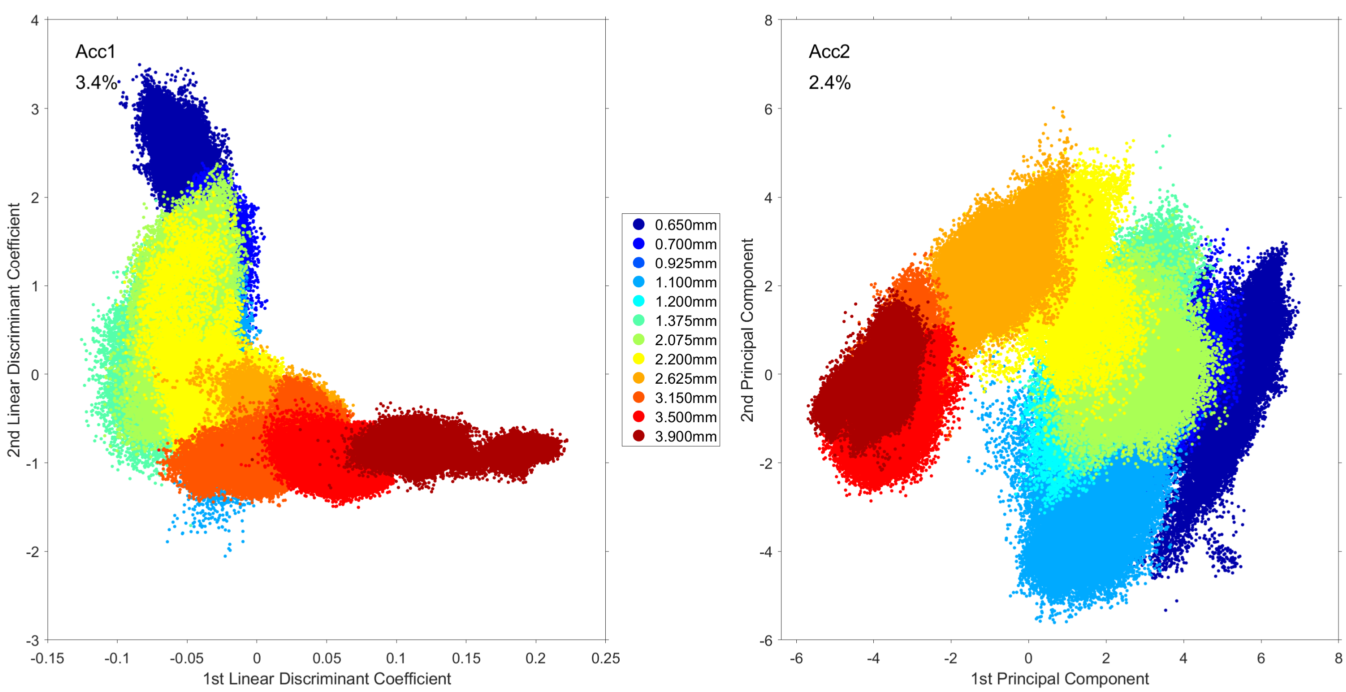

Figure 5. LDA (left) and PCA (right) of the features of the test set, with prediction coloured depending on the assigned class and accuracy displayed in the top left corner.

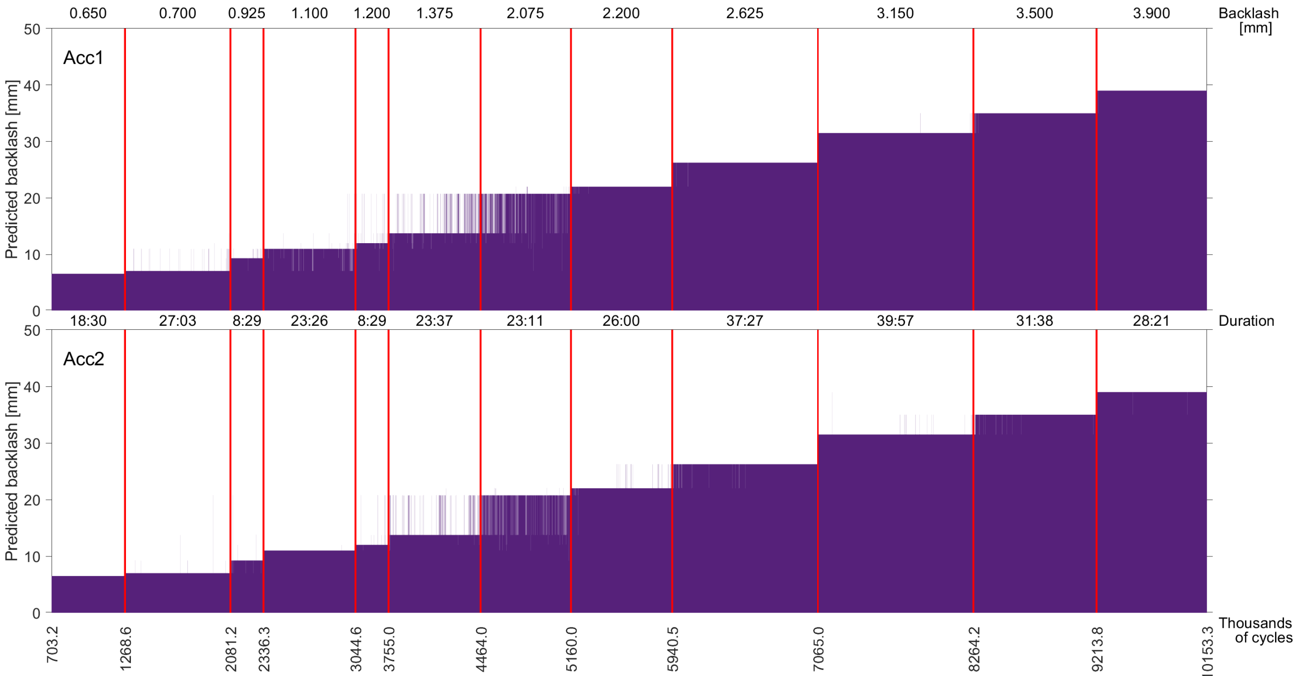

Figure 6. Label prediction in time. Red lines separate the different working phases. The top, middle and bottom horizontal axes display the backlash level, the working phase duration and number of machine cycles reached, respectively.

	Predicted labels [mm]											
Actual labels [mm]	**0.650**	**0.700**	**0.925**	**1.100**	**1.200**	**1.375**	**2.075**	**2.200**	**2.625**	**3.150**	**3.500**	**3.900**
0.650	28 244	0	0	0	0	0	0	0	0	0	0	0
0.700	0	39 789	18	705	21	22	9	0	0	0	0	0
0.925	0	37	12 350	330	0	0	0	0	0	0	0	0
1.100	2	1 122	203	33 690	178	113	76	5	0	0	0	0
1.200	0	15	0	274	11 764	467	203	1	0	0	0	0
1.375	0	13	1	249	268	30 363	4 521	2	0	0	0	0
2.075	0	29	0	73	282	4 825	29 346	207	0	0	0	0
2.200	0	0	0	0	0	0	306	38 651	6	0	0	0
2.625	0	0	0	0	0	0	0	33	56 124	0	0	0
3.150	0	0	0	0	0	0	0	0	0	59 808	96	0
3.500	0	0	0	0	0	0	0	0	0	261	47 121	8
3.900	0	0	0	0	0	0	0	0	0	0	18	42 447

Table 2. Confusion matrix of Acc1 test data. Each row contains the total number of models belonging to the relative class distributed in each column according to the predicted label.

	Predicted labels [mm]											
Actual labels [mm]	**0.650**	**0.700**	**0.925**	**1.100**	**1.200**	**1.375**	**2.075**	**2.200**	**2.625**	**3.150**	**3.500**	**3.900**
0.650	28 244	0	0	0	0	0	0	0	0	0	0	0
0.700	0	40 412	111	3	0	29	9	0	0	0	0	0
0.925	0	168	12 520	10	2	7	10	0	0	0	0	0
1.100	0	6	13	35 308	28	10	23	1	0	0	0	0
1.200	0	0	3	13	12 407	126	173	2	0	0	0	0
1.375	0	2	4	34	75	31 973	3 300	29	0	0	0	0
2.075	2	14	11	55	151	3 126	31 232	171	0	0	0	0
2.200	0	0	0	0	6	13	253	38 011	680	0	0	0
2.625	0	0	0	0	0	0	0	672	55 485	0	0	0
3.150	0	0	0	0	0	0	0	0	1	59 458	440	5
3.500	0	0	0	0	0	0	0	1	0	951	46 422	16
3.900	0	0	0	0	0	0	0	0	0	8	44	42 413

Table 3. Confusion matrix of Acc2 test data. Each row contains the total number of models belonging to the relative class distributed in each column according to the predicted label.

trend, similar to the backlash one. The IS indicator turns out to be useful for implementing fault detection policies, with thresholds applied to its level. However, it does not permit a clear diagnosis of the severity of the degradation. The same distance value corresponds to several backlash amounts, particularly in the early stages of mechanism deterioration. In this case, a suitable threshold for fault detection may be defined at around a level of 5 for Acc1 and 1 for Acc2.

Figure 5 and 6 depicts the results of the application of the SVM algorithm to predict labels of the models' test set for Acc1 (left) and Acc2 (right). In particular, Figure 5 shows the projection of that feature set using Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA), on the left and right side, respectively. Points color corresponds to the predicted labels assigned by the algorithm, with the achieved classification accuracy displayed in the top left corner of each plot. A value of 3.4% for Acc1 and 2.4% for Acc2 indicates positive results for PHM. Moreover, despite being point projections, it is possible to observe the path that features follow while the machine is degrading. On the other hand, Figure 6 outlines predicted labels against the actual ones, revealing an almost complete stairstep graph. There, it is possible to observe that the SVM predictor mostly struggles with backlash levels 1.375 mm and 2.075 mm when giving inaccurate outcomes, and notice that most of the misspredictions are around the class dividing line. Moreover, this can also be observed in the behaviour of the IS distance in Figure 4, where the index holds the same value for both the said classes. In this respect, we also provide the resulting confusion matrices in tables 2 and 3 for Acc1 and Acc2, respectively. Each row represents the test set of a single class, subdivided into the columns according to the predicted labels. Numbers in table rows sum up to the total models amount of the relative test set.

The SVM classifier results able to predict accurately the various degrees of wearing in the mechanism and the use of the two-level monitoring architecture is capable of providing information for predictive maintenance decision making. On the other hand, the IS distance results to be a valuable indicator for fault detection: Its computation on the controller does not require data exchange with external computers. However, it is advisable for the definition of preventive maintenance policies and not of predictive maintenance strategies.

5. CONCLUSION

In this paper, we studied the application of a PHM procedure that relies on the use of standard manufacturing computing solutions for its implementation, by analyzing its impact on a real-world case study, i.e., an industrial paper feeder. We achieve condition monitoring of its main mechanism backlash level with a distributed architecture. The PLC acts as an edge-computing unit and refines sensor measurements by exploiting the Model-of-Signal technique. Then, it sends them to a remote PC to outsource the final elaborations. The supervisor collects the received models and labels them with the backlash level recorded by the operator. At this point, the computer trains an SVM predictor and tests its classification accuracy. The reported results show that such predictor is a reliable solution for the generation of PHM indications. The management can use this information for the optimization of machinery components maintenance decisions. In this case, they can plan the servicing to bring the backlash level to normal.

ACKNOWLEDGMENTS

The authors would like to thank SITMA MACHINERY S.p.A. (`https://www.sitma.it/`) for supporting this project with the best-suited equipment and facilities and for providing insight and expertise that greatly assisted our work.

REFERENCES

Atamuradov, V., Medjaher, K., Dersin, P., Lamoureux, B., & Zerhouni, N. (2017). Prognostics and health management for maintenance practitioners-review, implementation and tools evaluation. *International Journal of Prognostics and Health Management*, *8*(060), 1–31.

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

Barbieri, M., Bosso, A., Conficoni, C., Diversi, R., Sartini, M., & 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* (pp. 263–269). Wiley – ISTE.

Barbieri, M., Diversi, R., & Tilli, A. (2019). Condition monitoring of ball bearings using estimated AR models as logistic regression features. In *Proc. of the 18th European Control Conference (ECC2019)* (pp. 3904–3909).

Barbieri, M., Diversi, R., & Tilli, A. (2020). Condition monitoring of electric-cam mechanisms based on model-of-signals of the drive current higher-order differences. In IFAC (Ed.), *Proc. of the 21st IFAC World Congress.*

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

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

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

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297.

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

Javed, K., Gouriveau, R., & Zerhouni, N. (2017). State of the art and taxonomy of prognostics approaches, trends of prognostics applications and open issues towards maturity at different technology readiness levels. *Mechanical Systems and Signal Processing*, *94*, 214–236.

Lee, J., Wu, F., Zhao, W., Ghaffari, M., Liao, L., & 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.

Söderström, T., & Stoica, P. (1989). *System identification*. Prentice Hall.

Vapnik, V. N. (Ed.). (1995). *The nature of statistical learning theory*. Springer.

Vogl, G. W., Weiss, B. A., & Helu, M. (2019). A review of diagnostic and prognostic capabilities and best practices for manufacturing. *Journal of Intelligent Manufacturing*, *30*(1), 79–95.

Wei, B., & Gibson, J. (2000). Comparison of distance measures in discrete spectral modeling. In *Proc. of the 9th Digital Signal Processing Workshop.*