Authors' accepted manuscript: Engineering Applications of Artificial Intelligence, 2020.
The final publication is available at: https://doi.org/10.1016/j.engappai.2019.103289

A Predictive Model for the Maintenance of Industrial Machinery in the Context of Industry 4.0

Jose-Raul Ruiz-Sarmiento[a,*], Javier Monroy[a], Francisco-Angel Moreno[a], Cipriano Galindo[a], Jose-Maria Bonelo[b], Javier Gonzalez-Jimenez[a]

[a]*Machine Perception and Intelligent Robotics Group (MAPIR), System Engineering and Automation Department, Biomedical Research Institute of Málaga (IBIMA), University of Málaga, Campus de Teatinos, 29071, Málaga, Spain.*
[b]*ACERINOX Europa S.A.U., Av. Acerinox Europa, 11379, Los Barrios, Cádiz, Spain.*

Abstract

The *Industry 4.0* paradigm is being increasingly adopted in the production, distribution and commercialization chains worldwide. The integration of the cutting-edge techniques behind it entails a deep and complex revolution –changing from scheduled-based processes to smart, reactive ones– that has to be thoroughly applied at different levels. Aiming to shed some light on the path towards such evolution, this work presents an Industry 4.0 based approach for facing a key aspect within factories: the health assessment of critical assets. This work is framed in the context of the innovative project *SiMoDiM*, which pursues the design and integration of a predictive maintenance system for the stainless steel industry. As a case of study, it focuses on the machinery involved in the production of high-quality steel sheets, i.e. the *Hot Rolling Process*, and concretely on predicting the degradation of the drums within the heating coilers of *Steckel mills* (parts with an expensive replacement that work under severe mechanical and thermal stresses). This paper describes a predictive model based on a *Bayesian Filter*, a tool from the *Machine Learning* field, to estimate and predict the gradual degradation of such machinery, permitting the operators to make informed decisions regarding maintenance operations. For achieving that, the proposed model iteratively fuses expert knowledge with real time information coming from the hot rolling processes carried out in the factory. The predictive model has been fitted and evaluated with real data from $\sim$118k processes, proving its virtues for promoting the Industry 4.0 era.

Keywords: Industry 4.0, Predictive maintenance, Machine Learning, Data Analysis, Smart manufacturing, Intelligent prognostics tools

1. Introduction

Initiatives towards *Industry 4.0* (*e.g. High-Tech Strategy 2020* [1], *Advanced Manufacturing Partnership* [2], or *Factories of the Future* [3]) share all the same goal [4]: to take advantage of the progresses in Cyber-Physical Systems (CPS) [5], Internet of Things (IoT) [6], Internet of Services (IoS) [7], and Big Data [8], for successfully facing the recent changes in economic, social, and environmental requirements for the manufacturing industries [9, 10, 11]. The recent developments in the aforementioned technological fields result in companies with networked assets producing a massive amount of data, coming in different formats and qualities [12]. A cornerstone inside the Industry 4.0 paradigm is the exploitation of such data in order to evolve from scheduled, control-based processes and systems to smart ones, able to predict the behaviour of the different actors involved in the industry value chain (*e.g.* customers, operators, machines, etc.) and anticipate them by self-adjusting their operations at different levels.

The key goal behind this paradigm is to boost the production efficiency of the modern industry, looking forward to achieve the commonly referred *self-aware*, *self-predict* and *self-maintain* abilities [13, 14]. However, an unavoidable fact that seriously threatens this goal is the occurrence of costly, unscheduled downtime and unexpected breakdowns [15, 16, 17]. The task in charge of palliating these situations is such of maintenance, which might involve the repair or replacement of components or parts and the disposal of damaged products [18]. The most basic maintenance approach is the so called *fail-and-fix* or *corrective maintenance*, which comes into play when the equipment fails and needs repair. This naive strategy evolved to the *preventive* or *blindly proactive* one, which according to past experiences, assesses a certain degradation profile for the machinery parts to plan ahead maintenance or replacing tasks according to their expected lifetime [19]. Thus, this *static* approach could end up with the replacement of parts still in good operating conditions, or with machine failures due to a faster degradation than expected, becoming both costly and ineffective actions.

In networked factories, new maintenance opportunities arise with the availability of massive data from processes and systems. This permits operators (or even intelligent scheduling systems) to monitor the machinery conditions instead of their

*Corresponding author
Email addresses: `jotaraul@uma.es` (Jose-Raul Ruiz-Sarmiento), `jgmonroy@uma.es` (Javier Monroy), `famoreno@uma.es` (Francisco-Angel Moreno), `cgalindo@uma.es` (Cipriano Galindo), `josemaria.bonelo@acerinox.com` (Jose-Maria Bonelo), `javiergonzalez@uma.es` (Javier Gonzalez-Jimenez)

Authors' accepted manuscript: Engineering Applications of Artificial Intelligence, 2020.
The final publication is available at: https://doi.org/10.1016/j.engappai.2019.103289

faults, hence anticipating possible failures, and optimizing the assets utilization. This advanced approach is commonly called *predictive maintenance* (PdM), and can be considered as an enhancement of preventive maintenance with *just-in-time* works. It relies on the assumption that the monitored machine parts go through a measurable process of degradation, hence enabling the estimation of temporal windows for carrying out preventive operations [20]. Predictive maintenance exhibits a number of inherent benefits, namely: optimized parts usage, reduced costs, increased machinery lifetime, plant safety, product quality (*near* zero failure manufacturing), reduced number of accidents, or effortless integration with company scheduling, among others [21].

Despite the benefits of PdM systems, their implementation in real factories remains challenging, mainly due to the required integration of different Industry 4.0-based technologies [22, 15]. Major challenges are: (i) the processing of large repositories of time series data from logistics, scheduling, and production (coming from noisy sensors mounted on the machinery) in order to be rendered in a usable form for their exploitation [23], and (ii) the design of a predictive model that, from such processed data, estimates the machinery condition in short time in order to perform an agile and informed decision-making. Such a model must be also able to learn from new data and to adapt its operation according to different situations [24]. PdM systems could also benefit from the consideration of information coming from experts regarding machines' operation, which is a valuable knowledge traditionally used in artificial intelligence-based solutions for industry. These issues need to be faced not only for PdM, but for most Industry 4.0-based solutions [13].

In this work, we contribute our solutions for facing the challenges of (i) data processing and rendering, and (ii) predictive model design, which have been deployed in a factory of *ACERINOX Europa S.A.U.* [25], a world class group in stainless steel production. Concretely, we present an intelligent predictive maintenance model based on the *Discrete Bayes Filter* [27, 28], one of the most widespread *Machine Learning* (ML) techniques [26]. We opted for Machine Learning as the tool of choice given the need to learn from unstructured data to make decisions, while the particular model to use (Support Vector Machines, Decision Trees, Artificial Neural Networks, etc.) depends on the peculiarities of the application at hand [26, 29, 30]. In this way, DBFs naturally address the challenges posed by PdM, namely: they (i) seamlessly integrate information coming from experts, configuration variables, and data from sensors, (ii) handle the uncertainty inherent to processes and noisy sensor measurements, (iii) fit their internal parameters to accommodate new information and react to new working conditions, and (iv) run in short time, enabling an agile decision making.

In a nutshell, the proposed DBF builds a probability distribution over a random variable x, modelling the part degradation condition, which can take discrete states whose values are commonly called *beliefs* [31]. These beliefs represent the uncertainty about the actual part degradation state, which can not be exactly retrieved due to the uncertainty coming from noisy sensors and processes, inaccurate models, etc. For example, if the part lifetime is discretized with values between 1 (new) and 10 (totally degraded), a computed belief of $Bel(x=1)=0.1$ indicates that it is unlikely that the part is new, while $Bel(x=7)=0.8$ means that it is quite probable that the part is seriously degraded. Every time that new process-related data are available, a new probability distribution is estimated and provided to the operators, along with a prediction of the part degradation after hypothetically carrying out certain process in the factory, enabling them to make informed decisions about the right time to replace the parts, or even to adapt the production. It is worth mentioning that our proposal goes beyond the usual operational limits of a laboratory-fitted model [24], since it is also able to learn from new data and to adapt its operation according to different conditions.

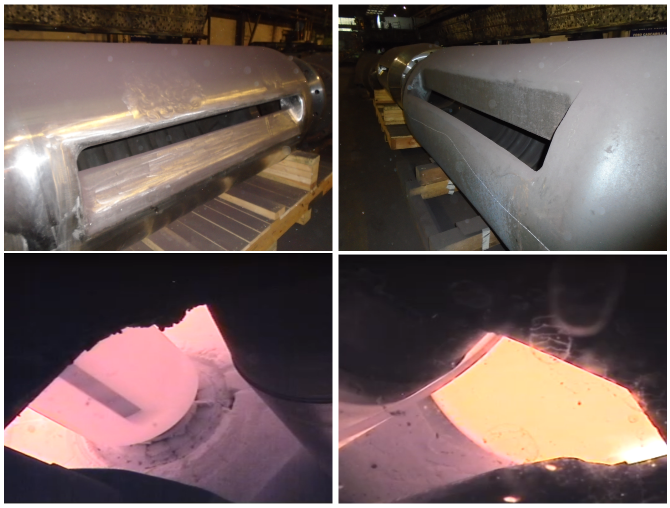


Figure 1: Top row, pictures of a new coiler drum (left) and a deteriorated one (right). Bottom row, pictures of a coiler drum while *working* within the Steckel mill: waiting for the steel sheet (left) and rolling it (right). The goal of the SiMoDiM project is to design a predictive system for making informed decisions regarding their just-in-time replacement.

This paper also addresses the challenge of rendering the massive data produced in the factory from logistics, scheduling and production into an usable form for their exploitation by the predictive model. For that, we provide recipes for analyzing and pre-processing these data, which were initially explored in our previous work [32], including: the acquisition of expert knowledge, descriptive and bivariate analyses of data, and a study of the influence of configuration variables.

Our work is carried out in the scope of the SiMoDiM project [33], which highlights a critical procedure within the stainless steel production: the multipass *Hot Rolling*, a mill process that involves rolling the steel at a high temperature to shape it. Specifically, the parts under degradation study are the drums within the coilers, which work under severe mechanical and thermal stresses and have an expensive replacement process. The top row of Fig. 1 shows two of these drums with different degradation states, while the bottom one shows the drums in action. The potential of the proposed PdM model has been assessed employing historical data from the hot rolling process carried out in the ACERINOX factory located in Cadiz, Spain. In that evaluation, its output is compared with the one from a preventive maintenance system, as well as with those from a number of traditional and state-of-the-art regression models, reporting more accurate predictions of the parts' degradation.

The next section puts our work in the context of the Industry 4.0 paradigm. Section 3 provides an overview of the full predictive maintenance system developed at SiMoDiM. Then, Section 5 introduces the proposed Bayes filter for predicting the machinery degradation, including a description of the available data as well as the proposed recipes for their processing. Section 6 reports the techniques used for validating the model and the obtained results. The paper concludes with a discussion about the work done in Section 7.

2. Related Work

This section starts presenting an overview of the main components and applications of Industry 4.0, including the maintenance of assets (see Section 2.1), to later describe the most common and widespread maintenance approaches, discuss popular frameworks for the implementation of predictive maintenance, and conclude relating our contribution to them.

2.1. Industry 4.0

A key principle of the Industry 4.0 paradigm is that processes and machinery must be networked as a collaborating community for the collection, exchange and analysis of data in order to predict future behaviours and pursue optimal solutions to possible problems [9, 11]. Nowadays, this principle is beginning to be achievable thanks to the development of a number of promising technologies. One of these technologies is the so-called *Cyber-Physical Systems* [5], which refers to systems with integrated computational and physical capabilities that can be interfaced in different ways [34, 35]. These systems are enhanced with features from the *Internet of Things* (IoT) [6] technology, providing them with the ability to continuously obtain information from sensors or processes across the factory, and securely forward it to (generally cloud-based) data centers [36]. This massive data production implies the development of new tools based on *Big Data* techniques [8, 14, 12], for storing, managing, and processing it. This set of technologies is completed with the *Internet of Services* (IoS) one, which takes the processed information from Big Data tools and deploys it at the right place and in the right form [7].

Within the Industry 4.0 paradigm, the number of works presenting models, frameworks, applications, use cases, etc., is rapidly growing, as reported by recent surveys [4, 10, 37, 38]. The aforementioned techniques enable a wide range of applications in different areas, *e.g.* process and planning (waste reduction and value increment), supply chain, transport and logistics, health and safety, product design, or the one addressed in this paper, maintenance and diagnosis. As an example, in the recommendations given in [1] the authors discuss their usage for: energy consumption reduction, end-to-end system engineering across the entire value chain, supporting custom manufacturing, telepresence, and sudden change of supplier during production. Energy management was also the application of discussion in the work by Shrouf *et al.* [39]. In the same way, Tamás and Illés [40] studied the trends in processes improvement that arose from this industrial revolution. Another interesting application is the creation of virtual worlds with augmented reality that, as described in [41], can be exploited for assisting operators in a dynamic production environment. This idea was further explored in the work by Simonis *et al.* [42]. Among this wide range of opportunities, this work tackles the application of Industry 4.0 principles for the predictive maintenance of assets, a critical aspect of companies' efficiency and products quality [43, 44, 45].

2.2. Predictive Maintenance

The maintenance of machinery in production lines is a task entailing significant factories' resources (budgets, operators, time, etc.) [13, 17]. The simplest maintenance option is *fail-and-fix* or *corrective maintenance* (also known as fire fighting), which essentially consists of repairing a machine when a failure is detected. This approach is prohibitive for modern factories, since unexpected breakdowns have a strong impact on companies' costs [21]. *Preventive maintenance* (PM) aims to avoid these breakdowns by scheduling periodical maintenance operations according to the elapsed time or machine usage. Although more appropriate, this approach has two disadvantages: it is also expensive to maintain (even more if the operation intervals are tight), and there is no learning about the machine degradation profile for future design improvements [46]. Precisely, the goal of the last evolution of this task, namely *predictive maintenance* (PdM), is to monitor the machinery status in order to decide when a repairment/replacement operation is needed given its degradation level. This approach has consensual benefits: cost reduction, operational efficiency, product quality improvement and increased flexibility [13]. Although the PdM paradigm is not new (it was introduced in late 1940s [22]), the emerging CPS, IoT, Big Data and IoS technologies have enabled its seamless application to modern industries [15].

A term closely related to PdM is such of Prognostics and Health Management (PHM), a discipline that pursuits the assessment of the machinery health and the prediction of its reliability and remaining useful lifetime (RUL) to carry out an informed system lifecycle management [47, 48]. As discussed in [49], PHM should not be considered a type of maintenance, but a set of techniques and methods that can be employed as maintenance inputs. Indeed, the predictive model proposed in this work could be considered a prognostic method.

There exist frameworks that pursue the formal definition of PdM systems. That is the case of the OSA-CBM architecture [50], which defines a standard for information flow to help to realize an end-to-end PdM system. A more practical example is SIMAP [51], a general tool that defines an architecture for the diagnosis and maintenance of industrial processes, which was applied to the retrieval of the health condition of a windturbine gearbox. Another popular framework is the Watchdog Agent®, of which [52] showed some implementation examples for online performance assessment: a commercial elevator door, a

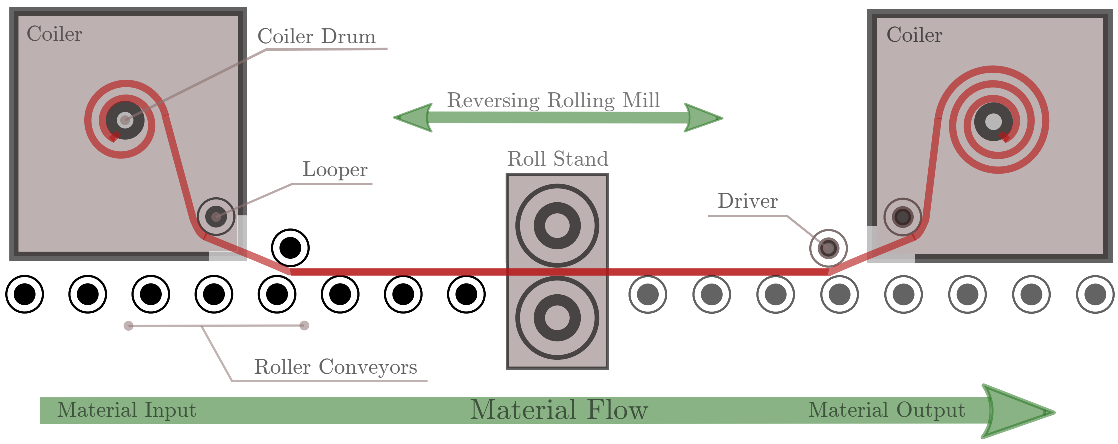


Figure 2: Scheme of the Steckel hot rolling mill. The stainless steel sheet (in red) is heated and worked in the mill through one or multiple passes, until desired thickness is obtained.

gear box in a material handling device, and different machine tools across a car manufacturing floor. The work by Gregor *et al.* [53] proposed a model to integrate PdM into the enterprise framework. Despite the OSA-CBM case, which defines an standard, the rest of proposals requires a significant effort for being adapted to each particular company requirements.

With the progressive application of PdM, the number of case studies that can be found in the literature is also increasing. For example, Lee *et al.* [21] described a cloud-based tool for the condition monitoring of a cutting machine, where its healthy status is reported through a web GUI to the operators. More related to our work we find that of Yang [54], where a Kalman filter is used to estimate the state of a DC motor for PdM purposes. As discussed in [22], this approach has the drawbacks of being complicated and computationally expensive, so it is not suitable for critical systems. The interested reader can find more case studies in the articles by Li *et al.* [16], Shin and Jun [55], Kesheng [56], and Prajapati *et al.* [22].

Those works rely on different techniques for determining the machinery state, most of them coming from the Artificial Intelligence (AI) or Machine Learning (ML) fields [57]. ML techniques are data-driven approaches able to find complex and non-linear patterns in data, and to build models from them that can be applied for prediction, detection, classification, or regression [29, 30, 17]. Among those techniques we can find Support Vector Machines, Decision Trees, Neural Networks, etc., being the selection of the model a matter of the: kind of data it has to work with, operational requirements, and type of results that it has to provide [52]. In the present paper we propose a PdM model that relies on a ML technique, a *Discrete Bayes Filter* [27] (DBF), for converting information from sensors, processes, and domain experts into knowledge about the machinery degradation state and its future behaviour. The information latent in this type of systems is naturally modelled by the filter, also providing a valuable measure of belief about its outcome. Another clear advantage of DBFs over other ML alternatives is their robustness against fluctuating and noisiy data. The proposed filter can also learn from experience, hence overcoming the tight test of time demanded by industrial settings [24]. Other filters like the Kalman filter [58] and its extensions [59] could be also considered, but they either rely on linearity assumptions of the underlying system, or require the calculus of complex Jacobians to linearly approximate its dynamics and to propagate uncertainty [60]. This work is carried out in the scope of the SiMoDiM project [33], with focus on the Steckel mills in the Hot Rolling process for the production of stainless steel [61]. To the best of our knowledge, this is the first work addressing the predictive maintenance of parts from a Steckel Mill.

3. Overview of the Predictive Maintenance System

Within the metallurgy industry, the purpose of the rolling process is to reduce and uniform the thickness of large pieces of metal (like semi-finished casting products, often called plates). These pieces are first heated over its recrystallization temperature (over 1700 °F [62]), and are then worked to reduce their thickness by passing one or several times through the mill. Fig. 2 shows a scheme of the Steckel hot rolling mill employed by ACERINOX Europa S.A.U. In it, the steel sheets run along the roller conveyors to be worked in the roll stand. If more than one pass is necessary, the metal sheets are coiled around the drum, and the process is repeated in the inverse direction (left-to-right, right-to-left). Due to the high temperatures of the process (the coilers contain a furnace to keep the steel temperature high) and the friction against the material, degradation of the machinery is common and a proper maintenance plan is mandatory to avoid costly and long production downtimes. This is the case of the coiler drums, the critical machinery parts we focus on, to assist in the decision of when they should be replaced/repaired [61].

Fig. 3 shows the proposed system for the predictive maintenance of those drums in the context of SiMoDiM. It starts by mounting on the mill a number of smart (embedded) sensors. These sensors are connected to a local network, converting the mill into a Cyber-Physical System (CPS), and are in

Authors' accepted manuscript: Engineering Applications of Artificial Intelligence, 2020.
The final publication is available at: https://doi.org/10.1016/j.engappai.2019.103289

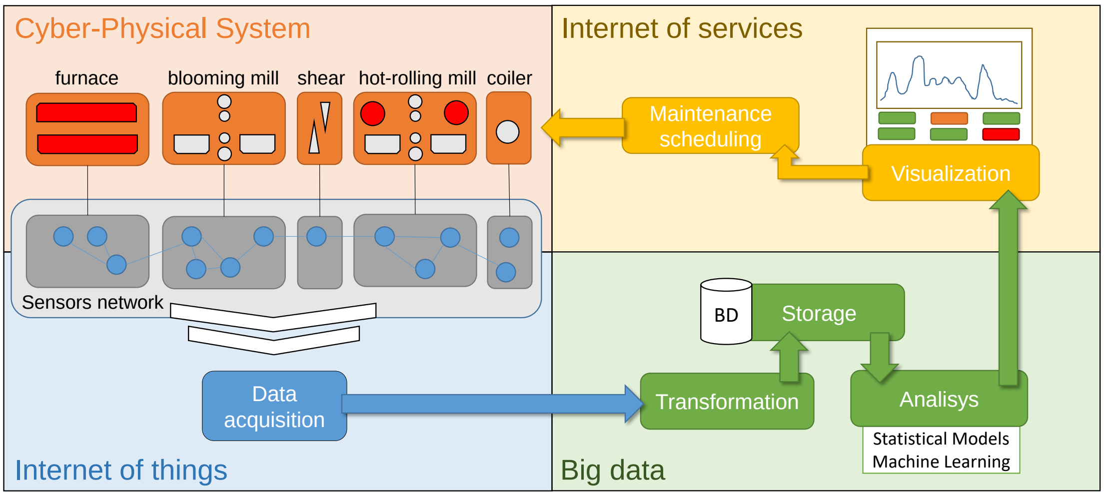


Figure 3: Logical components of the predictive maintenance system developed in SiMoDiM, using technologies from Cyber-Physical Systems, Internet of Things, Big data and Internet of Services. The Cyber-Physical Systems area shows, in addition to the hot-rolling-mill, the other components of the hot rolling pipeline.

charge of measuring parameters that change according to modifications and disturbances in the mill process. For each hot rolling process carried out in the mill, Internet of Things techniques are used to retrieve the collected information from the sensors, and to securely forward it to a centralized server. Recent advances in Big Data are then used to transform this networked data into a valid format for its storing into a NoSQL database, providing high capacity and fast query execution [63]. These data are analyzed and summarized by means of statistical tools (this is the data-to-information bridge [53]), and then a Machine Learning technique, concretely a Bayes Filter, is used to infer the degradation state of the coiler drums (information-to-knowledge bridge) as well as its future behaviour. Finally, through Internet of Services utilities, this knowledge is graphically shown to the operators in order to carry out an informed maintenance scheduling. This work focuses on two critical components within this system: (i) the analysis and processing of massive data, (ii) and the ML tool for predicting the machinery state, to which we refer as the *predictive model*. Next sections further describe these components. It is worth mentioning that, although we focus on the predictive maintenance of coiler drums, these components could be used on any other machinery part which degradation state can be observed, and with available ground truth information about its health over time.

4. Data analysis and processing

Our goal is to fit a ML model able to predict the evolution of the machinery degradation. In the ML framework, this can be achieved by means of a supervised learning process where the parameters are tuned according to pairs $(\mathbf{d}_i, gt_i)$ [13], with $\mathbf{d}_i$ representing the logistics, scheduling and production data from the hot rolling process p_i, and gt_i being the ground truth associated to that process, *i.e.* , the degradation state of the coilers. In general, it is better to have as much training data as possible. However, an excessive amount of them can hinder the detection of the relevant information and hence have a negative impact in the performance of ML techniques [24, 29]. Unfortunately, the massive data collected from CPSs are affected by this problem. To face this issue, it is required a *data processing* step in order to summarize $\mathbf{d}_i$, which is often called the data-to-information bridge, making it easier to find behaviour patterns about machinery degradation. For doing that, such data are characterized through representative *descriptors*. In this way, a descriptor can be defined as a simplified representation of some pieces of data that clarifies and preserves the relevant information for solving a certain problem. For example, in our case, a valid descriptor must carry information about the machinery health status.

Fig. 4 illustrates the design and working phases of the proposed ML model, where we can see how this data processing is conducted in the first two columns. The data gathered from each process carried out in the factory (first one) is summarized by useful descriptors (second column), resulting in a more concise, tractable and profitable information representation for fitting the model and, as described in Section 5, for inferring the machinery status.

To summarize, the purpose of the data processing step is to complete the data-to-information bridge, that is, to turn massive data into representative descriptors. However, how to accomplish this conversion is not a trivial task. We partially addressed this issue in our previous work [32] but, using such experience, in this section we provide concise recipes for successfully finding the most suited descriptors. Concretely, we propose a four-steps data analysis process in order to find the variables and descriptors that better summarize the raw data, as shown in Fig. 5. More formally, this analysis pursuits the design of the function $f(p_i)$ that renders the information coming from the process p_i into a vector of descriptors $\mathbf{d}_i = [d_{1i}, d_{2i}, \ldots, d_{ui}]$. The next sections introduce the proposed analysis steps, briefly (see Fig. 5):

1. *Acquisition of expert knowledge* (Section 4.2), which permits us to focus on promising logistics, scheduling or production information –also called *candidate variables*– according to the operational experience. It is also useful for

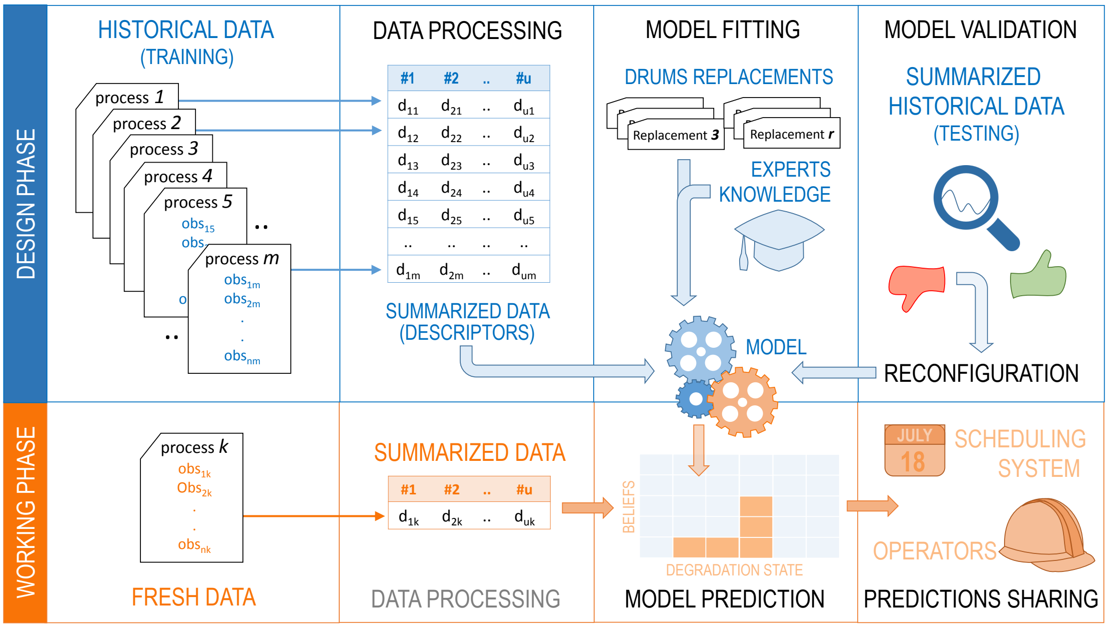


Figure 4: Diagram showing the information and processes involved in the design and working phases of the proposed predictive maintenance model. In the design phase, the production data from the factory are summarized through a number of descriptors that, along with experts knowledge and ground truth data, are used to tune the predictive model. In the working phase, fresh data coming from the factory are summarized in the same way and used to estimate the degradation state of the machinery and predict its behaviour in the future, information that is shared with the operators in order to schedule maintenance operations.

the identification of other factors that, in addition to the machinery status, could have influence in the variables values –called *interactions*– and have to be considered in the analysis. This step assists the remaining ones.

2. *Descriptive analysis* (Section 4.3), which pursues the study of the behaviour of the candidate variables and which descriptors could properly characterize them. For example, if a force is selected as a candidate variable, it could be characterized by its average, maximum and minimum values, its dispersion, amplitudes and frequencies in the frequency domain, etc.

3. *Bivariate analysis* (Section 4.4), which, once a number of descriptors have been proposed, validates them by trying to find patterns relating those descriptors with the machinery degradation status. If no relation is found for a given descriptor, then it is discarded.

4. *Analysis of the interactions with configuration variables* (Section 4.5), which aims to establish relations between configuration variables, *i.e.* those defining the behaviour of the process to be carried out (desired temperatures, features of the produced material, etc.), and the selected descriptors. If any interaction is found, that means that the descriptors' values are not only influenced by the machinery degradation, so they have to be considered in the prediction model.

For a better understanding of the analysis steps we exemplify them in the context of the SiMoDiM project, whose available data are described next.

4.1. The data

During the period 2013-2016, the data collected from the smart sensors mounted on the hot rolling machinery in the ACERINOX Europa S.A.U. factory in Cadiz, Spain, have been stored, building a rich and vast dataset. It contains information from the 118,484 hot rolling processes carried out in that period, each one available in the form of files that contain the value of 18 different variables measuring the processes' state at intervals of 0.5 meters of rolled steel (*e.g.* steel densities, coiler temperature, engines power or pressure and forces in the roll stand). Since the initial and desired thickness of the steel sheets differ, each file contains a different number of measurements or *rows*. This variability also results in hot rolling processes with different number of passes: 1, 3, 5, 7 and 9. As an illustrative example, the processes with 7 passes have between 5 and 6 thousand rows. In total, the number of provided measurements goes up to more than 436 millions (an average of $\sim$3.6k per process).

Each process is additionally identified by two *logistic variables*: number of campaign and steel plate identification, and 18 *meta-variables* or *configuration variables*. The latter consist of 8 variables reporting the properties of the steel plate (*e.g.* steel type, weight, length and thickness at the entrance, etc.), and 10 more configuring the behaviour of the process itself (work code, number of passes in the mill, coilers' temperature etc.). This sums up a total of 38 variables describing each hot rolling process. The resultant huge amount of data must be summarized using informative descriptors, process that is described in the next sections.

In addition to this, information regarding the dates where the coiler drums were replaced is also available, as well as monthly

Authors' accepted manuscript: Engineering Applications of Artificial Intelligence, 2020.
The final publication is available at: https://doi.org/10.1016/j.engappai.2019.103289

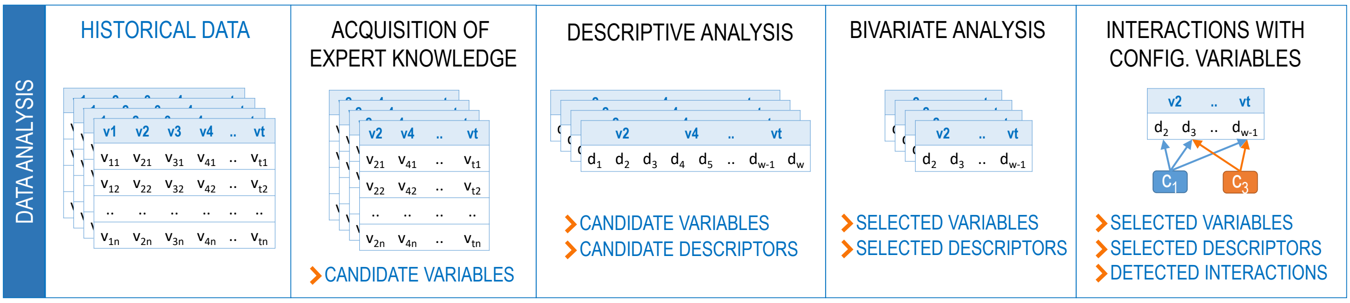


Figure 5: Diagram showing the steps of the proposed data analysis and their outcome, aiming to find the most appropriated features/descriptors to summarize the initial, raw data.

annotations about their health condition (ground truth). These replacement dates permit us to split the hot rolling processes into sequences, where each sequence spans over the lifetime of a coiler drum, ranging from 6 up to 8 months ($\sim$6.8 months on average). It is worth mentioning that, in these sequences, both coiler drums in the Steckel mill (recall Fig. 2) were replaced on the same date, hence sharing the same degradation profiles.

4.2. *Acquisition of expert knowledge*

To properly analyze the available data in order to find suitable descriptors, and avoiding turning the data analysis into an improvised and inaccurate process [57], it is necessary to obtain knowledge about the processes that produced the data [24], specially from experts on the domain. In the case of the SiMoDiM project, this knowledge was extracted through human elicitation. Next points summarize the retrieved information:

- **Clarification of candidate variables.** From the 18 variables describing the evolution of each hot-rolling process, two of them refer to the process progress (current pass and meters of steel processed), and other twelve reflect values that are consequence of the process configuration (aperture of the roll stand, temperature within the coilers, etc.). The remaining four are susceptible to being affected by the drums' state, hence being candidates to reflect their condition: *input* and *output-tension*, which measure the traction forces in both coilers, *leveling* that indicates the slope of the sheet being processed, and *bending* that measures its curvature. These variables are grouped into the vector $\mathbf{v}$.
- **Identification of possible interactions.** The experts from the factory pointed out that, in addition to other factors, the measurements of these variables are heavily influenced by the number of passes that the steel sheet does in the mill. To take this into account in the data analysis we used that parameter to cluster the processes, being the resultant groups individually studied, drastically reducing the data dispersion. They also highlighted the importance of looking for additional interactions, which is carried out in the last step of the proposed analysis (see Section 4.5).
- **Definition of parameters affecting the coilers' degradation.** The experts reported the factors that contribute the most to the parts' degradation. We make use of this valuable information for the prediction of the coilers' state after completing a hot rolling process, as described in Section 5.1.2. Concretely, these factors (coming from the configuration variables) are: the desired length of the steel sheet, the number of times that the material is rolled over the coilers, the type of steel, and the initial thickness of the plates.

This knowledge assists the analyses carried out in the next sections. At this point it is worth mentioning that the data analysis must be dynamic and not a one-time process, since the availability of new knowledge from experts, or evidences extracted from the data, would require its re-launching in order to reach more realistic conclusions.

4.3. *Descriptive analysis*

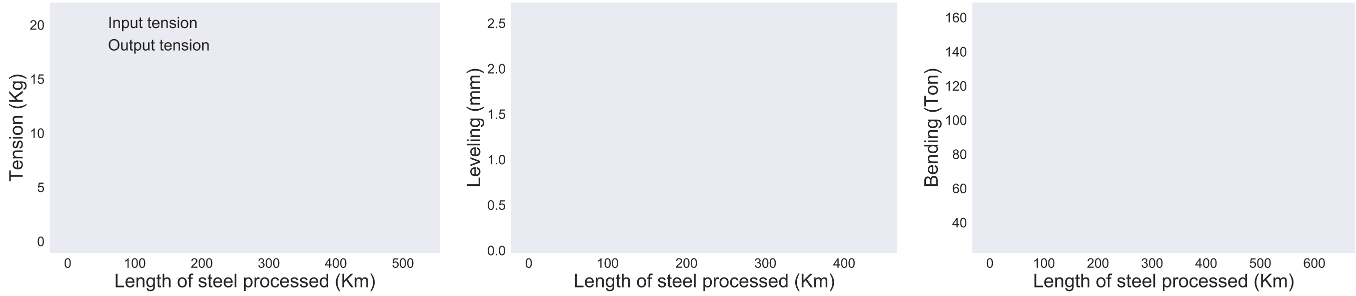
Analysis goal: to gain insight into the candidate variables $\mathbf{v}$, how they *behave* during the hot rolling processes, and which features or descriptors, represented by the vector $\mathbf{d}$, could be more suitable for their characterization.
Methods: First, the variables are described through their central tendency, dispersion and shape. This analysis is completed by graphical representations of their behaviour, as it is shown in Fig. 6. Notice that the measurements gathered by the sensors can be interpreted as temporal series or signals.
Results: according to the variables' behaviour, it was decided to extract features from the time domain for both *leveling* and *bending* variables, while for the case of the tensions they were also extracted features from the frequency domain (see Table 1).

Additionally, by means of this visual inspection it was also detected a strong correlation between the observed variables and the length of the sheets being processed (variable tightly related to the number of passes of the material in the mill). Fig. 6 illustrates this for the case of the *input-tension* and *output-tension* variables, where we can clearly see that the shorter the sheet, the higher the tensions, supporting the experts' intuitions in this respect.

4.4. *Bivariate analysis*

Analysis goal: to find patterns relating the behaviour of the characterized variables by the vector of descriptors $\mathbf{d}$, and the degradation state of the drums. This permits us to select the descriptors that best reflect the drums' condition.

Authors' accepted manuscript: Engineering Applications of Artificial Intelligence, 2020.
The final publication is available at: https://doi.org/10.1016/j.engappai.2019.103289



(a) (b) (c)

Figure 6: (a) Measured tensions from the 4^{th} pass in four 5-passes processes, (b) and (c), measured sheets' leveling and bending for different number of processes, respectively. The vertical, dotted lines indicate the processes' ending points.

Table 1: Computed descriptors for the different candidate variables (between parentheses, the number of total descriptors for each one). The descriptors of *output-tension* are the same as the *input-tension* ones. The check mark symbol (✓) identifies the final descriptors chosen to feed the predictive system.

Leveling	(3)
mean	1
hits	1
hill *stdv*	1

Bending	(3)
mean	1
hits	1
minimum	1

Input Ten.	(12)		
mean	1	Max. amp. FFT coefficients [✓]	3
stdv	1	Frequencies with max. amp.	3
hill mean	1	Skewness (frec. domain) [✓]	1
hill *stdv* [✓]	1	Kurtosis (frec. domain) [✓]	1

Methods: to numerically drive the analysis, we computed the well-known coefficient of determination [64], denoted as R^2, which represents the portion of the variance in a dependent variable (in this case, the descriptions of the candidate variables) that is predictable from an independent variable (drums degradation). This is a good measure about in which magnitude a variable influences another one, that is, how the degradation of the coilers affects the behaviour of the characterized variables. If there is no relation among them, then it would not be possible to predict the degradation using those descriptors. To compute it, first, a linear regression model is built, and then the model residuals are analyzed. The descriptors with $R^2 > 0.2$ and a reduced *p-value* (lower than 0.05) are chosen to comprise the final set of promising variables and descriptors.

Results: Table 1 shows a check mark next to the descriptors fulfilling this condition. Therefore, the vector of descriptors **d** describing each hot rolling process has 12 components (6 characterizing the input tensions and 6 for the output ones). Fig. 7 graphically illustrates this analysis for two variables: the sheet bending described through its mean values (per process), and the coiler input tensions by its hill oscillations (standard deviations). In both cases their values for two sequences (as well as their trends) are shown, and we can check that although the behaviour of the input tensions' oscillations is similar (both increase similarly, which means that the system becomes less stable with its degradation), it is not the case when analyzing the bending of the sheets.

4.5. Interaction with configuration variables

Analysis goal: to find out configuration variables correlated with the previously selected variables and descriptors. This is important to isolate the effect of the drums' state [21], and be able to better model the behaviour of the selected descriptors, which most probably not only depend on the coilers' conditions. One of these configuration variables has been already commented: the number of passes that the steel sheet does in the mill.

Methods: the coefficients of determination (R^2) between the candidate variables and the configuration ones were computed.

Results: The configuration variables showing a higher coefficient were the pass that the sheet is currently performing in the mill (first, second, third, etc., $R^2 = 0.25$), the width of the sheet before the hot rolling process ($R^2 = 0.20$), the target thickness after completing it ($R^2 = 0.20$), and the type of steel ($R^2 = 0.15$). This demonstrates the necessity of introducing these variables in the design step of the predictive model.

5. The Predictive Model

Once the massive data produced in the factory is rendered into a more concise and informative form, it can be used to fit a ML predictive model. This process is graphically illustrated in the third and fourth columns of Fig. 4 (top row). The chosen model relies on a Discrete Bayes Filter (DBF) [27]. This decision was made taking into account the challenges posed by predictive maintenance systems in the Industry 4.0 context, which are tackled by DBFs thanks to its capabilities for:

- Permitting us to seamlessly integrate data coming from logistics, scheduling, and production as well as knowledge from experts.
- Naturally handling the uncertainty inherent to sensor measurements, data transmission, inaccurate models, etc., which could lead to faulty systems if disregarded.
- Quickly adapting its behaviour to changes in the factory by fitting its internal parameters with the new data.
- Running in short time, providing fresh information to operators after each process carried out in the factory.

Authors' accepted manuscript: Engineering Applications of Artificial Intelligence, 2020.
The final publication is available at: https://doi.org/10.1016/j.engappai.2019.103289

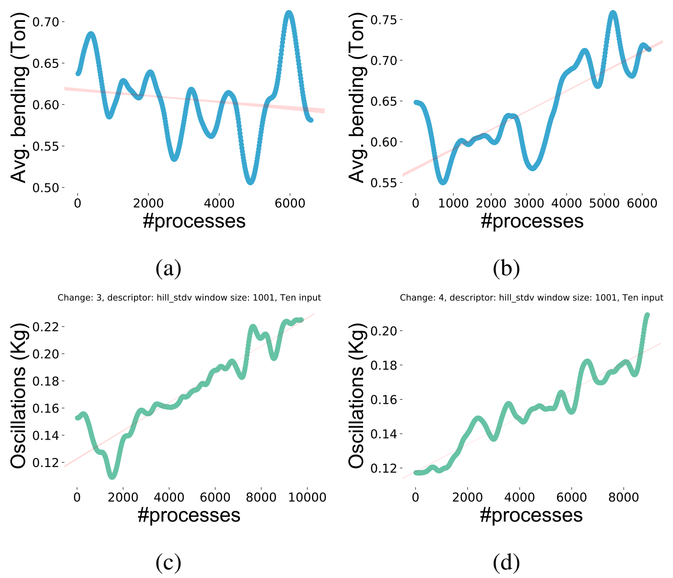


Figure 7: (a) and (b) illustrates the (per process) averaged sheets' bending for the second and third sequences, respectively, while (c) and (d) shows the oscillations of input tensions for the same sequences. The data have been smoothed to improve their visualization.

These and other features, like the compactness of DBFs, will be assessed throughout this section and later in the model evaluation one (see Section 6). In the following, Section 5.1 describes the core of the proposed Bayes Filter, while Section 5.2 discusses how to use it for estimating the machinery status and predicting its future behaviour, enabling operators to make informed decisions.

5.1. Discrete Bayes Filter Design

The *Bayesian Filters* seamlessly integrate the uncertain and limited knowledge about how the system works (in this case, the coiler drums), with the noisy measurements from the available sensors, aiming to make an estimation as accurate as possible of the system current state. In this project, the knowledge about the system evolution comes from the experts in the factory (recall the *Definition of parameters affecting the coilers degradation* point in Section 4.2), while the measurements are expressed as the processes' descriptors introduced in Section 4.

In this way, the *Discrete Bayes filter* (DBF) pursues the estimation of the degradation state of the coiler drums, considering for that a discrete range of values. Additionally, this estimation is enhanced with *beliefs* about the correctness of the inferred results. For example, let us consider that the health of the drums is discretized between the values 1 and 10 (meaning 1 a fresh drum, and a 10 a totally degraded one). Thereby, the estimation results in the beliefs about the current drum degradation belonging to each of these values (*e.g.* $Bel(x=1)=0.1$, $Bel(x=2)=0.45$, $Bel(x=3)=0.3$, etc.). In order to mathematically express the model, the following definitions are useful:

- N_s: number of values, or states, in which the drums' health/degradation is discretized.

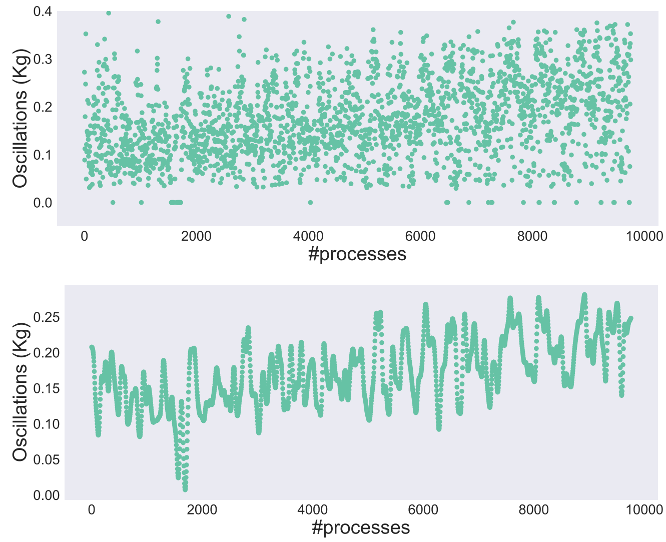


Figure 8: (Up) Oscillations of input tensions for the second sequence. (Bottom) Smoothed oscillations using a *Hanning* window of 101 processes.

- x: discrete random variable representing the drums' deterioration, taking values on the set $\{1,\dots,N_s\}$.
- k: represents a time instant, so x_k states the drum state at instant k.
- z_k: characterized measurements or observations of the sensors available at time k, that is, a reference to the vector of descriptors **d**.
- u_k: action taken on the system at time instant k, in this case a hot rolling process.
- c_k: configuration parameters of the process that have influence on the sensor measurements.

In the addressed problem, each time instant k corresponds to the instant in which a hot rolling process is completed in the mill. Thereby, we can retrieve the belief about the coiler drums having a certain degradation state after completing a process by computing:

$$Bel(x_k) = \eta\, P(z_k \mid x_k, c_k, z_{k-w:k-1}) \sum_{i=1}^{N_s} P(x_k \mid u_k, x^i_{k-1}) Bel(x^i_{k-1}) \quad (1)$$

where η is a normalization constant. We can see how the belief computation depends on two probability distributions, as well as on the belief at the previous instant, $Bel(x^i_{k-1})$. On the one hand, the distribution $P(z_k \mid x_k, c_k, z_{k-w:k-1})$ models the probability of having the observation (or descriptors' values) z_k given a certain drum degradation (x_k), the process configuration variables that have influence on it (c_k), and the previous measurements ($z_{k-w:k-1}$) within a window of processes of length w. This probability is often called *sensor model*.

On the other hand, the distribution $P(x_k \mid u_k, x^i_{k-1})$ models the behaviour or evolution of the system given a certain control action, that is, performing a hot rolling process. Concretely, it represents the probability of having a certain degradation state

Authors' accepted manuscript: Engineering Applications of Artificial Intelligence, 2020.
The final publication is available at: https://doi.org/10.1016/j.engappai.2019.103289

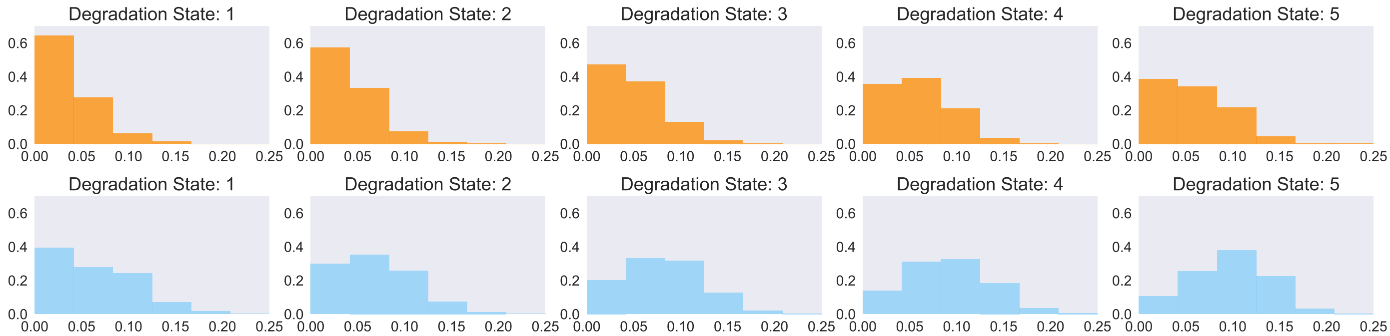


Figure 9: Histograms grouping the values of a descriptor (input tension oscillations). In this case, the drum degradation in discretized into 5 states. The top row includes values with an associated configuration variable *target thickness* lower than 4.25cm, while the second row includes those with a thickness higher than that value.

x_k given a control action (u_k) and a previous degradation state ($x_{k-1} = i$). This distribution also receives the name of *action model*. The next sections describe how these models have been defined (Section 5.1.1 and Section 5.1.2), while Section 5.2 introduces how to use them for estimating the machinery status and predict its future behaviour.

5.1.1. Sensor model

The sensor model specifies the probability to which sensors' measurements are generated given a deterioration state of the drum and a certain value of the configuration variables. Ideally, the sensors' observations depend on the current drum state, the values of such configuration variables, and an additional noise ξ coming from the sensors themselves, that is:

$$z_k = h(x_k, c_k) + \xi \qquad (2)$$

Given the noisy sensor readings, and hence noisy descriptors, (as illustrated in Fig. 8), and the influence that configuration variables (number of passes in the mill, desired output length, etc.) have on such readings, it is unfeasible to fit a sensor model employing only the current measurement z_k and ignoring such variables. Pursuing a more robust behaviour against these factors, we propose the design of a sensor model employing histograms of measurements.

For doing that, an histogram H is built for each possible degradation state N_s, and for each possible value of the configuration variables. For that, configuration variables taking continuous values are discretized. For example, the discretized values v_d for the configuration variable *desired thickness*, represented by v, can be retrieved as: $v_d = 0$ if $v < 4.25$, and $v_d = 1$ otherwise. The number of discretized values and their ranges are different for each configuration variable, and were set using expert knowledge and cross-validation. Notice that the total number of histograms to be built is $N_s \times \prod_{i=1}^{|c|} |v_d^i|$, being v_d^i the set of values that the configuration variable c_i can take, and $|\cdot|$ an operator returning the number of elements in a set/vector. Each histogram can be indexed as $H_{x,\mathbf{c}}$, where $x \in \{1, \dots, N_s\}$ represents a degradation state, and $\mathbf{c} = [v_d^1, \dots, v_d^{|c|}]$ states the discretized values of the configuration variables. For example, Fig. 9 shows the histograms built by discretizing the drum degradation into 5 states, and considering only the aforementioned configuration variable: the target thickness after the hot rolling process. The top row illustrates the 5 histograms grouping into 6 bins the measurements (input tension oscillations) from 4 sequences (see Section 6) that have a target thickness lower than 4.25, while the bottom row does the same for the measurements with a higher thickness. It can be clearly seen how, in each row, the distribution of the measurements differs, highlighting the need for using configuration variables in the design of the sensor model.

Thereby, in order to use these histograms within the sensor model, the last w measurements taken from the factory are used to build an additional histogram for each possible combination of configuration variables, $H_{z_{k-w:k},c}$. These histograms group together measurements that share the same configuration variables values. In this way, at time instant k, the sensor model computes the probability of the histogram $H_{z_{k-w:k},c_k}$ being produced by a machine with a degradation state x_k by comparing it with those histograms built for that state and configuration variables' values. Formally:

$$P(z_k \mid x_k, c_k, z_{k-w:k}) = \eta \left(1 - 0.5 dist(H_{x_k,c_k}, H_{z_{k-w:k},c_k})\right) \qquad (3)$$

where η is a normalization constant, and $dist(\cdot) \in [0,2]$ computes the *Mahalanobis* distance between the two histograms. It can be noticed that, if the two histograms are similar, the computed distance will be low, so the probability for the checked state will be high. Likewise, if they considerably differ, it would return a high distance and a low probability.

5.1.2. Action model

This model describes the behaviour or evolution of the system after the execution of an action, in this case, a hot rolling process. Thus, its purpose is to define at what extent the drum degrades given the execution of a process at instant k, denoted by u_k, which is described through a number of parameters q_k that influence on such degradation, for example, the number of times that the material is rolled over the coilers, the type of steel, the initial and desired thickness of the steel sheet, etc. (recall Section 4.2). This model is defined as follows:

Authors' accepted manuscript: Engineering Applications of Artificial Intelligence, 2020.
The final publication is available at: https://doi.org/10.1016/j.engappai.2019.103289

$$P(x_k \mid u_k, x_{k-1}) = \begin{cases} \sum_{i=1}^{|q|} l^i \times q_k^i \times N_s / \bar{q}_{tp}^i, & \text{if } x_k = x_{k-1}. \\ 1 - \sum_{i=1}^{|q|} l^i \times q_k^i \times N_s / \bar{q}_{tp}^i, & \text{if } x_k = x_{k-1}+1. \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

being l^i the normalized value for the influence of the process configuration parameter q^i, as given by the experts, so $\sum_{i=1}^{|q|} l^i = 1$, and $\bar{q}_{tp}$ the averaged sum of such configuration parameter in the sequences used to fit the model. For example, considering the variable *number of steps in the mill*, $\bar{q}_{tp}$ corresponds to the average of the total number of steps carried out by all the processes within each sequence.

This model sets that it is neither possible for the drum to regenerate ($x_k < x_{k-1}$), nor to degrade more than a discretized state unit ($x_k > x_{k-1}+1$). Therefore, for every new action u_k, this model estimates the probability of the drums to stay in the current degradation state ($x_k = x_{k-1}$), or to advance to the next discretized state ($x_k = x_{k-1}+1$).

5.2. *The Model in Action*

Once the parts of the DBF have been introduced, the next sections describe how to use them to estimate the degradation state of the coiler drums (Section 5.2.1), as well as how to predict its future behaviour (Section 5.2.2). Fig. 4 (bottom row) puts these steps in context with the working phase of the predictive maintenance model.

5.2.1. *Estimation of Machinery Condition*

The DBF principle is that the state of the system at instant k can be estimated from the state at instant $k-1$, the control action u_k carried out, and the taken measurements z_k. For that, two steps are employed: prediction and update.

- In the *prediction step* the beliefs at the previous instant $Bel(x_{k-1})$ are refreshed using the action model. That is, the filter *predicts* the new belief about the machinery having a certain degradation after carrying out a hot rolling process. It corresponds with the summation in Eq. (1), and it adds uncertainty to the inference (it spreads the belief over the different system states). In this way, only using the predictions to estimate the drum condition could end up having no information about it (the higher the number of predictions, the higher the uncertainty about the state). That is why it is needed to consider additional information for achieving a reliable degradation estimation.

- At this point is where the *update step* comes to play, taking advantage of the sensor model and the taken measurements. Such an update corresponds with the first part of Eq. (1), and permits the filter to increase/decrease the belief about the drum being at certain states, decreasing in this way the uncertainty about the inference results.

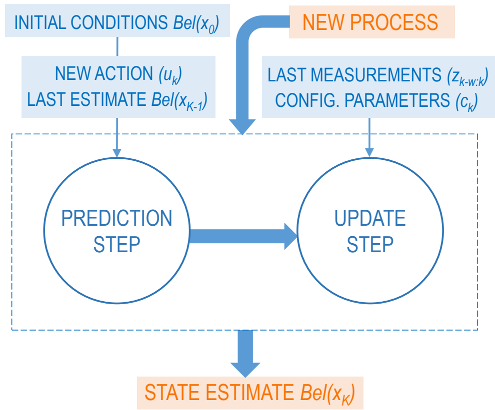


Figure 10: Diagram of the Discrete Bayes Filter workflow for estimating the degradation of the machinery each time that a new hot rolling process is carried out.

Fig. 10 outlines the filter operation each time that a hot rolling process is completed in the factory, while Alg. 1 provides a more formal algorithmic description. Before the first execution of the filter (which is launched each time a hot rolling process is completed), a number of initial conditions are set, with $Bel(x_1 = 1) = 1$ and $Bel(x_i = 1) = 0, \forall i \neq 1$. Such initial conditions just say that the coiler drums are new ($Bel(x_1 = 1) = 1$). Then, the algorithm starts by entering into a loop, where it predicts the state at instant k according to the beliefs at instant $k-1$, the configuration parameters of the rolling process completed at the mill, and the action model (lines 2-3 in Alg. 1, prediction step). After this, another loop updates the previous prediction with the measurements taken by the sensors and the sensor model, which computes which state is more probable that the measurements belong to (lines 4-9, update step). The result of the update step, after normalization, is some intermediate point between the prediction and the previous similarity computation, hence becoming the estimation $Bel(x_k)$ (lines 8-10).

For a better understanding of the filter outcome, let us discuss some possible results. Fig. 11 (top row) shows two estimations of the coiler drums degradation after completing 6k processes, illustrating two possible scenarios in the factory. In the first one (left), the output of the model tells the operators that it is fairly confident about the degradation being at state 9, so the coiler drums can still be used, while in the second one (right) the estimation is not such confident, so it should be collected additional information to determine its state, or scheduled a maintenance task in order to avoid a fail.

5.2.2. *Prediction of Future Behaviour*

Up to this point we have described a model able to estimate the health status of the machinery. However, for doing predictive maintenance it can also be useful to provide a prediction about the system behaviour in the future. For doing so, after having estimated the degradation of the machinery at a time instant k in the form of beliefs $Bel(x_k)$, the DBF can be used to predict its degradation at time instant $t+n_p$, that is, after hypothetically performing n_p processes. For doing so they are used

Authors' accepted manuscript: Engineering Applications of Artificial Intelligence, 2020.
The final publication is available at: https://doi.org/10.1016/j.engappai.2019.103289

Algorithm 1 Discrete Bayes Filter iteration

procedure DBF(
  $Bel(x_{k-1})$, ▷ *Belief at previous time instant*
  u_k, ▷ *Action taken on the system*
  $z_{k-w:k}$, ▷ *Measurements (last w processes)*
  c_k, ▷ *Process configuration parameters.*
)
  for all x_k **do** ▷ *Prediction step*
    $Bel(x_k) = \sum_{i=1}^{N_s} P(x_k \mid u_k, x_{k-1}^i) Bel(x_{k-1}^i)$
  $\eta = 0$
  for all x_k **do** ▷ *Update step*
    $Bel(x_k) = P(z_k \mid x_k, c_k, z_{k-w:k-1}) Bel(x_k)$
    $\eta = \eta + Bel(x_k)$
  for all x_k **do**
    $Bel(x_k) = \eta^{-1} Bel(x_k)$
  return $(Bel(x_k))$

Algorithm 2 Discrete Bayes Filter Prediction

1: **procedure** DBF_PREDICTION(
  $Bel(x_k)$, ▷ *Estimated belief at time instant k*
  $\mathbf{u}$, ▷ *Future actions to be executed*
  n_p ▷ *Number of future actions*
)
2: **for** $j \in \{1 \cdots n_p\}$ **do**
3:    $Bel(x_{k+j}) = \sum_{i=1}^{N_s} P(x_{k+j} \mid u_{k+j}, x_{k+j-1}^i) Bel(x_{k+j-1}^i)$
4: **for all** x_{k+n_p} **do**
5:    $Bel(x_{k+n_p}) = Bel(x_{k+n_p}) / \sum_{i=1}^{N_s} Bel(x_{k+n_p} = i)$
6: **return** $(Bel(x_{k+n_p}))$

the action model as well as the n_p future control actions of the hot rolling processes to be completed.

Alg. 2 details how to perform such prediction. Briefly, the action model is used to iteratively compute the predicted beliefs after hypothetically carrying out each hot rolling process (lines 2-3), then the beliefs after the last process are normalized to sum up 1 (lines 4-5), retrieving the beliefs $Bel(x_{k+n_p})$ that predict the degradation state of the coiler drums after carrying out those hot rolling processes (line 6). In this way, factory operators are provided with valuable information for carrying out just-in-time maintenance operations.

Let us illustrate how these predictions could be used and interpreted by the operators. For example, suppose that a pair of coiler drums have completed 6k processes, and their degradation state is estimated by the beliefs provided in Fig. 11 (top-left). Then, the operators can employ this tool to predict their degradation after, for example, other additional 500 processes. For that, they feed the system with the configuration parameters for such $u_{k+1} \cdots u_{k+500}$ processes, and execute the model to obtain the predictions. Fig. 11 exemplifies the model output in two different scenarios: (bottom-left) after carrying out 500 process not so demanding for the coiler drums (low number of passes, a favorable steel type, etc.), and (bottom-right) after completing 500 demanding manufacturing processes. In the first scenario, the operators can infer that the coiler drums still have enough health to keep working after performing those processes, although in the second one the execution of those processes will require the scheduling of maintenance operations after their completion to prevent undesired breakdowns. The factory, as a consequence of these results and its production target, could re-schedule the configuration of the hot rolling processes to be carried out in the factory, *i.e.* to execute processes with more or less demanding profiles, enabling it to accordingly lengthen or shorten the life span of the machinery.

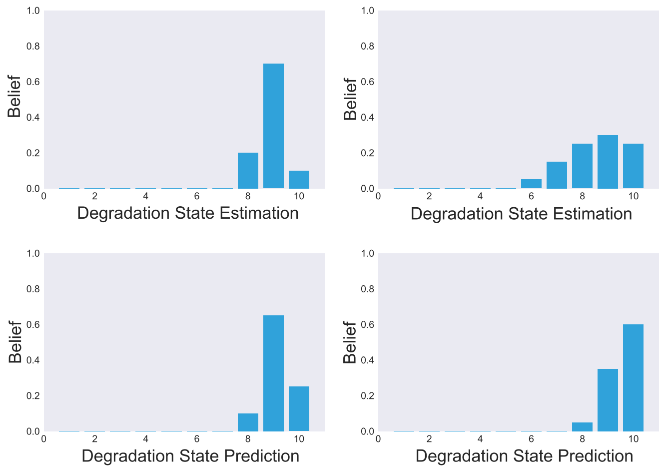


Figure 11: Top, two examples of estimations of the degradation state of the coiler drums after carrying out 6k hot rolling processes. Bottom, two examples of predictions of such degradation after hypothetically completing 500 processes more (having the top-left estimation as the initial belief).

6. Model Evaluation

We have employed the data provided by ACERINOX Europa S.A.U. , as described in Section 4.1, to evaluate the performance of the predictive model. The outcome of this study is addressed in Section 6.1, while Section 6.2 analyses how the decisions made during the sensor model design (*e.g.* size of the window of processes, recall Section 5.1.1) influence such performance, and Section 6.3 explores the learning capabilities of the model, an important feature for systems operating in industrial settings.

6.1. Analysis of Model Performance

We have employed the well known *cross-validation* technique [65] for carrying out the model evaluation, following a *leave-one-out* approach [23]. In a nutshell, from the five available sequences of processes in the dataset, four of them are chosen for fitting the model, while the remaining one is used for evaluation. This process is repeated changing the sequence used for evaluation, completing in total five training-testing cycles. The model performance is retrieved by averaging the outcome of each test.

Table 2: Performance of the proposed DBF-based model (predictive maintenance) versus a solution only employing knowledge from experts (preventive maintenance), as well as traditional and state-of-the-art ML regression models, measured according to the reported Root Mean Squared Error (bold numbers highlight the better results).

Model	Total RMSE	RMSE avg.	RMSE stdv.
Knowledge-based	3.84	0.77	0.47
DBF-based	**2.98**	**0.59**	**0.10**
OLS	3.67	0.73	0.29
Nearest Neighbors	6.41	1.28	0.79
Decision Trees	4.92	0.98	0.28
MLP	3.74	0.75	0.27
AdaBoost	4.15	0.83	0.34
l-SVM	3.64	0.73	0.13

To measure the performance, we have employed the Root Mean Squared Error (RMSE) reported by the model predictions, that is:

$$RMSE = \sqrt{\frac{1}{N_p}\sum_{i=1}^{N_p}(y_i - \hat{y}_i)^2} \tag{5}$$

where N_p is the total number of hot rolling processes in the testing sequence, y_i is the ground truth drum degradation, and $\hat{y}_i$ is the degradation estimated by the model. While ground truth information is continuous, the output of the DBF is a vector of beliefs over discrete states, so for retrieving a continuous value for such a degradation at each time step we used a *weighted average*:

$$\bar{y}_i = \sum_{j=1}^{N_s} j\,Bel(x_i = j) \tag{6}$$

6.1.1. Comparison with a purely preventive approach

First, the performance of the proposed DBF-model is going to be compared with the one from a purely preventive approach, that is, a Knowledge-Based model. Such model only employs the knowledge acquired from experts to estimate the drum state, *i.e.* it only performs prediction steps (recall Section 5.1.2 and Section 5.2.1), so it is a form of preventive maintenance. In that comparison, shown in the two first rows of Table 2, the proposed model achieves a high performance, considerably reducing the total (sum of the RMSE reported by each validation sequence) and averaged RMSE in a ∼22% and a ∼23% respectively, and reaching a reduced standard deviation, that is, showing a similar behaviour independently of the sequence being used for testing. The last point is of special interest, and clearly states the added value of using a predictive model. Unlike preventive approaches, this model is able to detect the peculiarities of each sequence of processes instead of performing according to preset profiles.

These numbers have been obtained by discretizing the coiler drums' state into 10 values. As commented in Section 4.1, the coiler drums share degradation profiles, so they also exhibit similar degradation conditions. There are other parameters of the proposed model (*e.g.* the size of the window of processes,

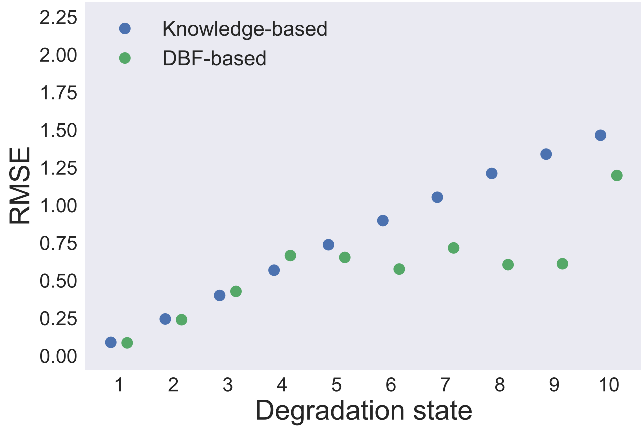


Figure 12: Evolution of the RMSE for the different degradation states of the coiler drums for the two compared approaches.

the number of bins in the histograms, etc.) that are fixed to certain values. These values were not handcrafted, but tuned following a cross-validation approach. Next sections go deeper into this fitting.

To further analyze the differences between the models in the spotlight, Fig. 12 reports their behaviours depending on the *ground truth* drum degradation state. We can see how they behave similar in the 5 first states, which indicates that the drums' degradations are similar for all the sequences in that period, but they clearly differentiate from then on. In the states 6, 7, 8 and 9, the DBF-based model is able to keep low the RMSE, and even to decrease the standard deviation of the errors, hence yielding more stable predictions over time. The RMSE increases in the last state, although it is still lower than the one reported by the Knowledge-based method. However, since the predictive maintenance operations must be carried out before the drum totally degrades, in practice this behaviour is not a drawback.

Another interesting study supporting these conclusions is reported in Fig. 13. This series of figures illustrates the evolution of the beliefs about the degradation state of a coiler drum when the ground truth degradation is 2.5, 5, 7.5 and 10, from left to right. On the one hand, the upper row represents this evolution for the Knowledge-based model, and we can see how the beliefs are increasingly spread over the states, that is, each time that the filter is executed it is less clear which degradation state is the right one. On the other hand, the bottom row reports the beliefs' evolution for the DBF-based model, and we can see how the inclusion of the update step and the sensor model helps to concentrate the beliefs and provide more accurate predictions.

6.1.2. Comparison with ML approaches

Additionally to the previous study, the performance of the DFB-based model has been also compared with those from other traditional and state-of-the-art ML regression models, which pursue the estimation of the machinery status according to found patterns among variables. Concretely, to conduct this analysis, we have relied on the scikit-learn [66] implementation

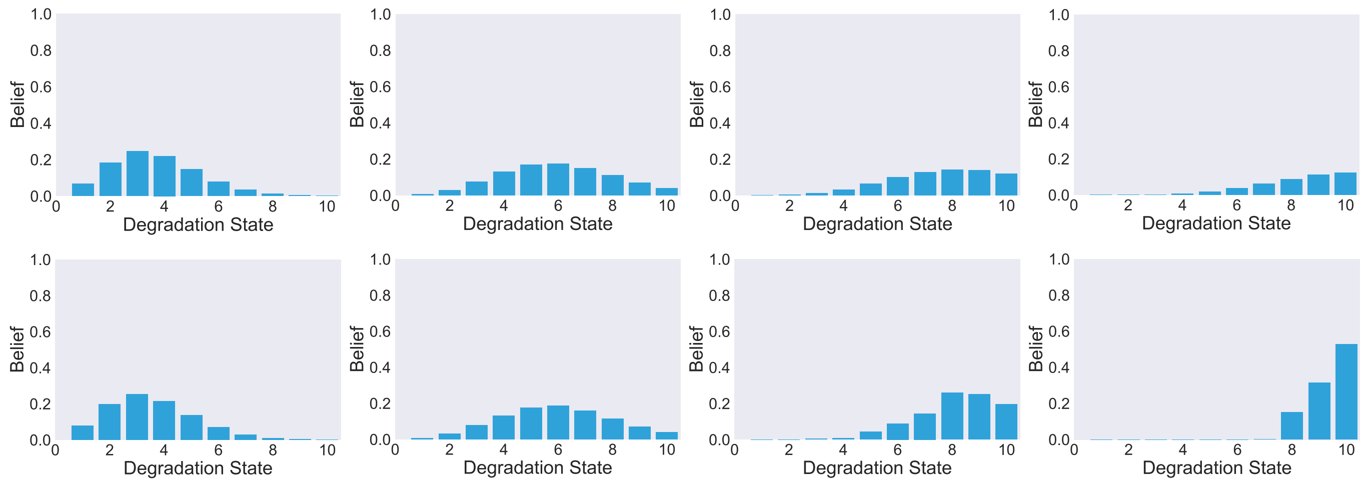


Figure 13: Evolution of the beliefs reported by the Knowledge-based (top row) and the DBF-based (bottom row) models at four different instants in the lifetime of a drum. That is, from left to right, a degradation of 25%, 50%, 75%, and 100%.

of Ordinary Least Squares (OLS), Nearest Neighbours, Decision Trees, Multi-layer Perceptron (MLP), AdaBoost and linear Support Vector Machines (l-SVM). The internal, configuration parameters of those models (e.g. number of hidden layers in the MLP, maximum depth of Decision Trees, etc.) have been set by cross-validation. The obtained results are reported in Table 2, where it can be seen that the proposed DBF-model reaches the highest success both in estimating the coiler drums degradation state (RMSE average) and in the stability of the results (RMSE standard deviation). This can be due to the robustness of DBFs against noisy and fluctuating data, and to how they seamlessly integrate expert knowledge and production data.

6.2. *Influence of the Sensor Model Design*

The effect of the window size. As commented in Section 5.1.1, the characterized measurements coming from the sensors are noisy [67], so their immediate utilization in the sensor model would end up with an *unstable* prediction. To face that, we proposed the utilization of a time-window to feed the model, so the descriptions from the last w processes are considered. To set this parameter, we resorted to cross-validation. In this way, we also include in this iterative process the window size w along with the variable specifying the sequence used to test the predictive model.

Fig. 14 reports the obtained RMSE by employing different window sizes. We can see how the best results regarding total and averaged RMSE are obtained for $w = 100$. Although we can achieve a more stable model performance by increasing this size, as shown by the obtained RMSE standard deviation, this leads to higher prediction errors.

Number of histogram bins. Other important factors when building the sensor model are the histograms that model its behaviour. The principal parameter to set in their design is the number of bins in which the observations are going to be grouped. Again, this value has been obtained through cross-validation, as reported by Fig. 15, where we can clearly see that the best results are obtained using 2 bins to group the measurements descriptors.

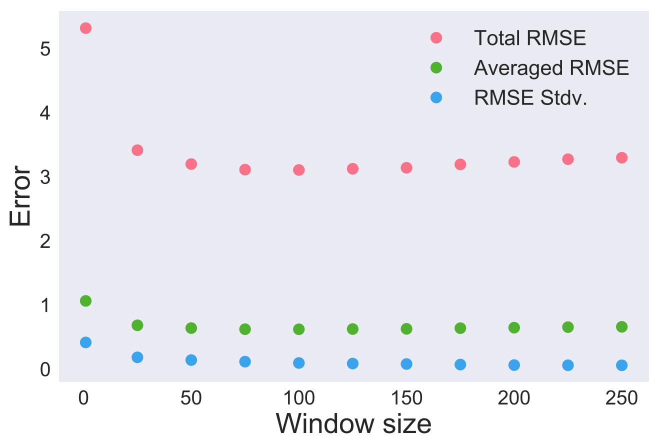


Figure 14: Evolution of the reported RMSE by the DBF-based model using windows of processes of different length.

6.3. *Evaluation of the Model Learning Capabilities*

A requirement for predictive maintenance systems working in industrial settings is to be able to learn from new data, as well as to show a certain grade of adaptability against changes in the processes [13, 24]. As discussed in [29]: *intelligence is strongly connected with learning, and learning ability must be an indispensable feature of Intelligent Manufacturing Systems*. To evaluate this, we have carried out diverse cross-validation processes changing the number of sequences used for fitting the model. In doing so, we are emulating a model that has to work with only one sequence available to train its parameters, then a new sequence is also available for fitting, and so on.

Table 3 provides the outcome of this study. We can see how the model is able to learn from new data, increasing its performance in a ~29% using 4 training sequences w.r.t. the case with only one. In fact, each time that a new sequence is available the averaged RMSE is reduced. Looking at this table we can also conclude that any DBF-based model with two or more training sequences has a better performance than the Knowledge-based

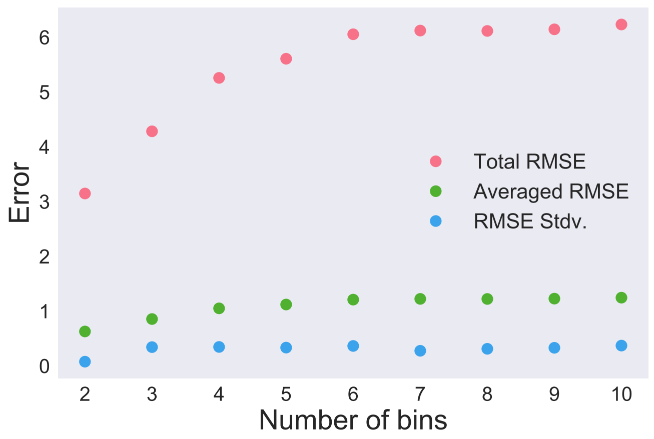


Figure 15: Evolution of the reported RMSE by the DBF-based model using histograms with different number of bins.

Table 3: Performance of the proposed DBF-based model according to the number of sequences used for its training.

#Training seq.	1	2	3	4
RMSE avg.	0.83	0.70	0.64	0.59
RMSE stdv.	0.16	0.09	0.07	0.1

one (recall Table 2).

These results suggest that the addition of even more data from sequences of processes would further improve the performance, but it could not be the case if the processes in the factory change (*e.g.* they use new materials, the coiler drum manufacturer changes, etc.). A way to provide an advanced capacity of adaptation to the model could be to weight the training data depending on its date of collection, having the newest sequences a higher relevance.

6.4. *Analysis of the Model Complexity*

This section analyses the complexity of the model in terms of the number of parameters needed to define it, and shows how it is able to run in short time. The parameters of the DBF are included in the sensor and action models. On the one hand, the sensor model needs $\prod_{i=1}^{|c|} N_s \times |v_d^i| \times b^i$ parameters, being c the set of configuration variables, N_s the number of possible degradation states, $|v_d^i|$ the number of values in which the configuration variable c_i is discretized, and b^i the number of bins in the histograms for that variable. On the other hand, the action model needs $2 \times |q|$, being q the configuration variables that influence the coiler drum degradation. In our model instance we needed 488 parameters in total, 480 for the sensor model and 8 for the action model. If they are represented by floating-point numbers, with an usual size of 4 bytes, then our model total size would be of $\sim 2kB$, resulting in a lightweight and compact model that could be deployed even in systems with restrictive memory-size constraints.

Regarding its execution time, on average, the update step of the algorithm (recall Alg. 1) needs $1.4ms$, while the prediction step takes $0.05ms$. This short execution time enables its utilization for predicting the machinery behaviour in the future as, for example, its state after hypothetically carrying out 1000 process could be computed in only $50ms$. Finally, concerning the time needed to fit the model, it depends on the number of sequences used for that, but it is also short. Concretely, the training with one sequence needed $3.37s$, $7.40s$ for two, $11.46s$ with three sequences, and $14.50s$ for four of them.

7. Conclusions and Future Work

This work has described an Industry 4.0 based solution for the health assessment of the machinery within factories. Concretely, the machinery under study are the coiler drums within Steckel mills, critical components in the hot rolling process for the production of sheets of stainless steel. These parts have a costly replacement, requiring a smart scheduling of maintenance operations in order to keep the plant efficiency high. For achieving so, it is proposed a predictive model as part of a prognosis system that takes advantage of the core technologies behind the Industry 4.0 paradigm. The workflow of such system is as follows: sensors mounted on the machinery (Cyber-Pysical Systems) produce data that are networked (Internet of Things), stored and managed (Big Data) to inform the operators about the machinery state (Internet of Services). Beyond this general description, the paper has gone into details on how to render the collected sensory data into an usable form for the predictive model supporting such maintenance. In this way, they have been provided recipes for processing such data, including the acquisition of expert knowledge, how to carry out descriptive and bivariate analyses, and how to find relations among sensor measurements and configuration variables (those setting the behaviour of the hot rolling processes).

Once the data are in an usable form, we propose its exploitation by a Discrete Bayes Filter (DBF), which plays the role of a predictive model that estimates the machinery condition and is able to predict its behaviour after the execution of a number of processes, assisting operators to make informed decisions regarding scheduling. This model iteratively fuses valuable sources of information like expert knowledge, configuration parameters, and real time information coming from sensors mounted on the machines. The performance of the predictive model has been assessed with real data from a factory in Cádiz, Spain, owned by ACERINOX Europa S.A.U. , demonstrating its virtues. In that evaluation, the DBF-based model was compared with a Knowledge-Based one, typically used for carrying out a preventive maintenance, as well as with traditional and state-of-the-art Machine Learning regression models, showing an improved operation of the proposed model. The model compactness and efficiency have been analyzed, reporting a reduced size in memory and short execution times. It has been also shown how to fit the most relevant DBF parameters for obtaining a high performance. Finally, the ability of the model for learning from new data as been assessed in an emulated ongoing evaluation over time. The obtained results are certainly promising towards the Industry 4.0 era.

In the future we plan to keep receiving data from the ACERINOX Europa S.A.U. factory, and use them to further evaluate the robustness of the method against changes in the manufacturing process (new or different production materials, coiler drums of different brands, etc.). A promising way to handle this issue is the weighting of the training data, giving more relevance to the newest sequences.

Acknowledgements: Work partially funded by the projects SiMoDiM ([OTRI-8.06/5.56.4826], [IC4-030000-2016-3]), and WISER ([DPI2014-55826-R]), financed by the Spanish Ministry of Industry, Energy and Tourism, and the Ministry of Economy, Industry and Competitiveness respectively, and by a postdoc contract from the I-PPIT-UMA program financed by the University of Málaga.

References

[1] H. Kagermann, W. Wahlster, J. Helbig, Recommendations for implementing the strategic initiative industrie 4.0 – securing the future of german manufacturing industry, Final report of the industrie 4.0 working group, acatech – National Academy of Science and Engineering, München (April 2013).
URL http://forschungsunion.de/pdf/industrie_4_0_final_report.pdf

[2] R. Reif, S. A. Jackson, A. Liveris, Report To The President Accelerating U.S. Advanced Manufacturing, Washington, dc: The presidents council of advisors on science and technology (2014).
URL https://www.broadinstitute.org/files/sections/about/PCAST/2014%20amp20_report_final.pdf

[3] European Commission, Factories of the future PPP: towards competitive EU manufacturing, Research and innovation, european union (2013).
URL http://ec.europa.eu/research/press/2013/pdf/ppp/fof_factsheet.pdf

[4] Y. Liao, F. Deschamps, E. de Freitas Rocha Loures, L. F. P. Ramos, Past, present and future of industry 4.0 - a systematic literature review and research agenda proposal, International Journal of Production Research 55 (12) (2017) 3609–3629. doi:10.1080/00207543.2017.1308576.
URL https://doi.org/10.1080/00207543.2017.1308576

[5] R. Baheti, H. Gill, Cyber-physical systems, The impact of control technology 12 (2011) 161–166.

[6] F. Xia, L. T. Yang, L. Wang, A. Vinel, Internet of things, International Journal of Communication Systems 25 (9) (2012) 1101.

[7] L. Atzori, A. Iera, G. Morabito, The internet of things: A survey, Computer networks 54 (15) (2010) 2787–2805.

[8] N. Khan, I. Yaqoob, I. Hashem, Z. Inayat, W. Kamaleldin, M. Alam, M. Shiraz, A. Gani, Big data: Survey, technologies, opportunities, and challenges 2014 (2014) 18.

[9] H. Lasi, P. Fettke, H.-G. Kemper, T. Feld, M. Hoffmann, Industry 4.0, Business & Information Systems Engineering 6 (4) (2014) 239–242.

[10] R. Sreedharan.V, A. Unnikrishnan, Moving towards industry 4.0: A systematic review, International Journal of Production Research 117 (20) (2017) 929–936.

[11] H. S. Kang, J. Y. Lee, S. Choi, H. Kim, J. H. Park, J. Y. Son, B. H. Kim, S. Do Noh, Smart manufacturing: Past research, present findings, and future directions, International Journal of Precision Engineering and Manufacturing-Green Technology 3 (1) (2016) 111–128.

[12] M. Khan, X. Wu, X. Xu, W. Dou, Big data challenges and opportunities in the hype of industry 4.0, in: 2017 IEEE International Conference on Communications (ICC), 2017, pp. 1–6.

[13] B. Nikolic, J. Ignjatic, N. Suzic, B. Stevanov, A. Rikalovic, Predictive manufacturing systems in industry 4.0: Trends, benefits and challenges, in: Proceedings of the 28th DAAAM International Symposium, 2017, pp. 796–802.

[14] P. ODonovan, K. Leahy, K. Bruton, D. T. OSullivan, Big data in manufacturing: a systematic mapping study, Journal of Big Data 2 (1) (2015) 20.

[15] J. Lee, H.-A. Kao, S. Yang, Service innovation and smart analytics for industry 4.0 and big data environment, Procedia CIRP 16 (2014) 3–8. doi:https://doi.org/10.1016/j.procir.2014.02.001.

[16] Z. Li, K. Wang, Y. He, Industry 4.0 – potentials for predictive maintenance, in: International Workshop of Advanced Manufacturing and Automation, 2016, pp. 42–46.

[17] N. Amruthnath, T. Gupta, A research study on unsupervised machine learning algorithms for early fault detection in predictive maintenance, in: 5th International Conference on Industrial Engineering and Applications, 2018.

[18] European Committee for Standardization, Maintenance - maintenance terminology, in 13306 european standard (2017).

[19] H. Erbe, G. Morel, J. Lee, B. Iung, J. Lger, G. Seliger, E. Hohwieler, D. Kiritsis, M. Hecht, U. Berger, Infotronic technologies for e-maintenance regarding the cost aspects, IFAC Proceedings Volumes 38 (1) (2005) 1 – 12, 16th IFAC World Congress. doi:https://doi.org/10.3182/20050703-6-CZ-1902.01384.
URL http://www.sciencedirect.com/science/article/pii/S1474667016373967

[20] R. K. Mobley, An introduction to predictive maintenance, Butterworth-Heinemann, 2002.

[21] J. Lee, E. Lapira, S. Yang, A. Kao, Predictive manufacturing system - trends of next-generation production systems, IFAC Proceedings Volumes 46 (7) (2013) 150 – 156, 11th IFAC Workshop on Intelligent Manufacturing Systems. doi:https://doi.org/10.3182/20130522-3-BR-4036.00107.
URL http://www.sciencedirect.com/science/article/pii/S1474667015356664

[22] A. Prajapati, J. Bechtel, S. Ganesan, Condition based maintenance: a survey, Journal of Quality in Maintenance Engineering 18 (4) (2012) 384–400. doi:10.1108/13552511211281552.
URL https://doi.org/10.1108/13552511211281552

[23] J. R. Ruiz-Sarmiento, C. Galindo, J. Gonzalez-Jimenez, Building multiversal semantic maps for mobile robot operation, Knowledge-Based Systems 119 (2017) 257–272. doi:10.1016/j.knosys.2016.12.016.

[24] H. Roger W., S. Ronald D., D. V. Richard D., Applying statistical thinking to Big Data problems, Wiley Interdisciplinary Reviews: Computational Statistics 6 (4) (2014) 222–232. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/wics.1306, doi:10.1002/wics.1306.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/wics.1306

[25] Acerinox Europe S.A.U. website, http://www.acerinox.com, [Online; accessed 01-October-2019] (2019).

[26] E. Alpaydin, Introduction to machine learning, MIT press, 2009.

[27] S. Särkkä, Bayesian filtering and smoothing, Vol. 3, Cambridge University Press, 2013.

[28] J.-L. Blanco, J. Monroy, J. Gonzalez-Jimenez, A. Lilienthal, A kalman filter based approach to probabilistic gas distribution mapping, in: Proceedings of the 28th Annual ACM Symposium on Applied Computing, 2013, pp. 217–222. doi:10.1145/2480362.2480409.
URL http://mapir.isa.uma.es/mapirwebsite/index.php/mapir-downloads/papers/205

[29] T. Wuest, D. Weimer, C. Irgens, K.-D. Thoben, Machine learning in manufacturing: advantages, challenges, and applications, Production & Manufacturing Research 4 (1) (2016) 23–45. doi:10.1080/21693277.2016.1192517.

[30] D. Wu, C. Jennings, J. Terpenny, R. X. Gao, S. Kumara, A comparative study on machine learning algorithms for smart manufacturing: tool wear prediction using random forests, Journal of Manufacturing Science and Engineering 139 (7) (2017) 071018.

[31] V. Fox, J. Hightower, L. Liao, D. Schulz, G. Borriello, Bayesian filtering for location estimation, IEEE Pervasive Computing 2 (3) (2003) 24–33. doi:10.1109/MPRV.2003.1228524.

[32] J. R. Ruiz-Sarmiento, J. Monroy, F.-A. Moreno, J. M. Bonelo, J. Gonzalez-Jimenez, Analysis of Data from the Industrial Machinery within the Hot Rolling Process for Predictive Maintenance, in: Frontiers of Artificial Intelligence and Applications, 2018. doi:10.3233/978-1-61499-929-4-122.

[33] SiMoDiM website, http://mapir.isa.uma.es/work/simodim, [Online; accessed 01-October-2019] (2019).

[34] L. Monostori, Cyber-physical production systems: roots, expectations and r&d challenges, Procedia CIRP 17 (2014), pp. 9-13.

[35] J. Lee, B. Bagheri, H.-A. Kao, A cyber-physical systems architecture for industry 4.0-based manufacturing systems, Manufacturing Letters 3

(2015) 18–23.

[36] J. Wan, S. Tang, Z. Shu, D. Li, S. Wang, M. Imran, A. V. Vasilakos, Software-defined industrial internet of things in the context of industry 4.0, IEEE Sensors Journal 16 (20) (2016) 7373–7380. doi:10.1109/JSEN.2016.2565621.

[37] Y. Lu, Industry 4.0: A survey on technologies, applications and open research issues, Journal of Industrial Information Integration 6 (2017) 1 – 10. doi:https://doi.org/10.1016/j.jii.2017.04.005. URL http://www.sciencedirect.com/science/article/pii/S2452414X17300043

[38] M. Brettel, N. Friederichsen, M. Keller, M. Rosenberg, How virtualization, decentralization and network building change the manufacturing landscape: An industry 4.0 perspective, International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering 8 (1) (2014) 37 – 44. URL http://waset.org/Publications?p=85

[39] F. Shrouf, J. Ordieres, G. Miragliotta, Smart factories in industry 4.0: A review of the concept and of energy management approached in production based on the internet of things paradigm, in: 2014 IEEE International Conference on Industrial Engineering and Engineering Management, 2014, pp. 697–701. doi:10.1109/IEEM.2014.7058728.

[40] P. Tamás, B. Illés, Process improvement trends for manufacturing systems in industry 4.0, Academic Journal of Manufacturing Engineering 14 (2016) 7.

[41] V. Paelke, Augmented reality in the smart factory: Supporting workers in an industry 4.0. environment, in: Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA), 2014, pp. 1–4. doi:10.1109/ETFA.2014.7005252.

[42] K. Simonis, Y.-S. Gloy, , T. Gries, Industrie 4.0 - automation in weft knitting technology, IOP Conference Series: Materials Science and Engineering 141 (2016) 012014. doi:10.1088/1757-899X/141/1/012014.

[43] G. Kabir, S. Tesfamariam, J. Loeppky, R. Sadiq, Predicting water main failures: A bayesian model updating approach, Knowledge-Based Systems 110 (2016) 144 – 156. doi:https://doi.org/10.1016/j.knosys.2016.07.024. URL http://www.sciencedirect.com/science/article/pii/S0950705116302386

[44] G. Medina-Oliva, A. Voisin, M. Monnin, J.-B. Leger, Predictive diagnosis based on a fleet-wide ontology approach, Knowledge-Based Systems 68 (2014) 40 – 57, enhancing Experience Reuse and Learning. doi:https://doi.org/10.1016/j.knosys.2013.12.020. URL http://www.sciencedirect.com/science/article/pii/S0950705113004000

[45] M.-H. Karray, B. Chebel-Morello, N. Zerhouni, Petra: Process evolution using a trace-based system on a maintenance platform, Knowledge-Based Systems 68 (2014) 21 – 39, enhancing Experience Reuse and Learning. doi:https://doi.org/10.1016/j.knosys.2014.03.010. URL http://www.sciencedirect.com/science/article/pii/S0950705114000938

[46] J. Lee, E. Lapira, Predictive factories: The next transformation, Manufacturing Leadership Journal 4 (1) (2013) 26 – 30, 11th IFAC Workshop on Intelligent Manufacturing Systems. doi:https://doi.org/10.3182/20130522-3-BR-4036.00107. URL http://www.sciencedirect.com/science/article/pii/S1474667015356664

[47] R. Gouriveau, K. Medjaher, N. Zerhouni, From prognostics and health systems management to predictive maintenance 1: monitoring and prognostics, John Wiley & Sons, 2016.

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

[49] A. C. Márquez, V. G.-P. Díaz, J. F. G. Fernández, Advanced maintenance modelling for asset management. Springer International Publishing, 2018. doi:https://doi.org/10.1007/978-3-319-58045-6.

[50] M. Lebold, K. Reichard, C. S. Byington, R. Orsagh, OSA-CBM architecture development with emphasis on xml implementations, in: The Maintenance and Reliability Conference, 2002.

[51] M. C. Garcia, M. A. Sanz-Bobi, J. del Pico, Simap: Intelligent system for predictive maintenance: Application to the health condition monitoring of a windturbine gearbox, Computers in Industry 57 (6) (2006) 552 – 568, e-maintenance Special Issue. doi:https://doi.org/10.1016/j.compind.2006.02.011. URL http://www.sciencedirect.com/science/article/pii/S0166361506000534

[52] D. Djurdjanovic, J. Lee, J. Ni, Watchdog agent–an infotronics-based prognostics approach for product performance degradation assessment and prediction, Advanced Engineering Informatics 17 (3) (2003) 109 – 125, intelligent Maintenance Systems. doi:https://doi.org/10.1016/j.aei.2004.07.005. URL http://www.sciencedirect.com/science/article/pii/S1474034604000102

[53] M. Gregor, M. Haluka, M. Fusko, P. Grznr, Model of intelligent maintenance systems, in: 26th DAAAM Internacional Symposium on Intelligent Manufacturing and Automation, 2016, pp. 1097–1101.

[54] S. Yang, An experiment of state estimation for predictive maintenance using kalman filter on a dc motor, Reliability Engineering & System Safety 75 (1) (2002) 103 – 111. doi:https://doi.org/10.1016/S0951-8320(01)00107-7. URL http://www.sciencedirect.com/science/article/pii/S0951832001001077

[55] J.-H. Shin, H.-B. Jun, On condition based maintenance policy, Journal of Computational Design and Engineering 2 (2) (2015) 119 – 127. doi:https://doi.org/10.1016/j.jcde.2014.12.006. URL http://www.sciencedirect.com/science/article/pii/S2288430014000141

[56] K. Wang, Intelligent predictive maintenance (ipdm) system – industry 4.0 scenario, WIT Transactions on Engineering Sciences 113 (1) (216) 259–268.

[57] M. Dopico, A. Gomez, D. De la Fuente, N. García, R. Rosillo, J. Puche, A vision of industry 4.0 from an artificial intelligence point of view, in: Proceedings on the International Conference on Artificial Intelligence (ICAI), The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2016, p. 407.

[58] G. Welch, G. Bishop, An introduction to the kalman filter, Tech. rep., Chapel Hill, NC, USA (1995).

[59] S. J. Julier, J. K. Uhlmann, New extension of the kalman filter to nonlinear systems, in: Signal processing, sensor fusion, and target recognition VI, Vol. 3068, International Society for Optics and Photonics, 1997, pp. 182–193.

[60] D. C. Swanson, A general prognostic tracking algorithm for predictive maintenance, in: 2001 IEEE Aerospace Conference Proceedings (Cat. No. 01TH8542), Vol. 6, IEEE, 2001, pp. 2971–2977.

[61] Y. V. Konovalov, A. S. Khokhlov, Benefits of steckel mills in rolling, Steel in Translation 43 (4) (2013) 206–211. doi:10.3103/S0967091213040062.

[62] M. Supermarkets, Difference between hot rolled steel and cold rolled steel, www.metalsupermarkets.com/difference-between-hot-rolled-steel-and-cold-rolled-steel/, [Online; accessed 01-October-2019] (2019).

[63] J. Pokorny, Nosql databases: a step to database scalability in web environment, International Journal of Web Information Systems 9 (1) (2013) 69–82.

[64] N. R. Draper, H. Smith, Applied regression analysis, Statistics in Medicine 19 (22) (1998) 3136–3139.

[65] S. Arlot, A. Celisse, et al., A survey of cross-validation procedures for model selection, Statistics surveys 4 (2010) 40–79.

[66] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[67] J. R. Ruiz-Sarmiento, D. Zuñiga-Noël, J. Gonzalez-Jimenez, Intrinsic Calibration of Depth Cameras for Mobile Robots Using a Radial Laser Scanner, International Conference on Computer Analysis of Images and Patterns. Springer, Cham, 2019. p. 659-671. doi:10.1007/978-3-030-29888-3_54.