Mechanical Systems and Signal Processing 210 (2024) 111163

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

journal homepage: www.elsevier.com/locate/ymssp

Hybrid prognostics to estimate cutting inserts remaining useful life based on direct wear observation

Luca Bernini [a,b,*], Ugo Malguzzi [b], Paolo Albertelli [a,b], Michele Monno [a,b]

[a] *Politecnico di Milano, Department of Mechanical Engineering, via La Masa, 1, Milan, 20156, Lombardy, Italy*
[b] *MUSP Macchine Utensili Sistemi di Produzione, strada della Torre della Razza, Piacenza, 29122, Emilia-Romagna, Italy*

ARTICLE INFO

Communicated by E. Chatzi

Keywords:
Tool wear
Hybrid adaptive prognostics
Turning
Particle filter
Neural networks
Variable cutting parameters

ABSTRACT

The aim of this study is to monitor tool wear through recurrent direct observation, to automatically and optimally assess when it is really necessary to change tools. To achieve this goal, a hybrid prognosis algorithm is formulated to estimate cutting tools' remaining useful life. Since cutting speed, and more generally process parameters, influences the rate of tool degradation, an adaptive prognosis strategy is presented on the basis of flank wear measurements through the application of a Particle Filter (PF) framework. The adaptability feature allows tracking changes in flank wear evolution. The idea is to fit available degradation curves of cutting tool flank land measurements, through the use of data-driven models, i.e. Multi-Layer Perceptrons (MLP) and cubic polynomials (P3). The Remaining Useful Life of the cutting tool is estimated together with its probability density function, by using a PF framework to adapt MLP weights (or P3 coefficients) along with online flank wear measurements. The devised algorithm was proven to adapt to wear trends from the field, obtained with cutting parameters not previously tested, making it suitable for a robust implementation. The approach was tested when trained upon a single run-to-failure, and validated upon four run-to-failures in different cutting conditions according to a cross-validation inspired technique. P3 was found to be more reliable (from the metrics perspective), whereas MLP allowed to be accurate with greater advance, offering a practical advantage. The proposed algorithm may also be adapted to integrate physical features, like specific force coefficients, with direct wear measurements.

1. Introduction

Digital manufacturing represents the synergy of production processes and data acquisition systems [1,2]. Acquired data can be employed to support different phases of the production process. Prognostics and Health Management (PHM) is one of these. PHM consists in the detection, diagnosis and prognosis of machine faults [3–5]. Anomaly detection consists in the identification of abnormal states, i.e. deviations from normal operating conditions; diagnosis reflects in the classification of such deviations, performing the isolation and quantification of the abnormal states; at last, prognosis performs the prediction of abnormal states evolution up to a total failure [4].

Indeed, the main goal of prognosis is to predict the Remaining Useful Life (*RUL*) of faulty components, as well as the estimation of its probability density function (PDF) and thus, its uncertainty bounds [6]. The prediction of faults is beneficial both in terms of cost savings on materials and tools, but most importantly it allows decreasing maintenance time. When dealing with cutting tools,

[*] Corresponding author at: Politecnico di Milano, Department of Mechanical Engineering, via La Masa, 1, Milan, 20156, Lombardy, Italy.
E-mail addresses: luca1.bernini@polimi.it (L. Bernini), ugo.malguzzi@musp.net (U. Malguzzi), paolo.albertelli@polimi.it (P. Albertelli), michele.monno@polimi.it (M. Monno).

https://doi.org/10.1016/j.ymssp.2024.111163
Received 28 January 2023; Received in revised form 13 January 2024; Accepted 16 January 2024
Available online 23 January 2024

the estimation of failure before their occurrences may end up in saving up to 40% of maintenance costs [7] and about 20% of downtimes of machine tools [8,9], responsible for production losses.

The focus of this paper is on the prognostics of cutting tools for turning operations. Turning is a widespread technology that allows producing axial symmetric solids of rotation. The turning process is a subtractive manufacturing technique which has been widely studied and modelled in the last decades. However, the possibility of developing adaptive strategies to estimate the remaining useful life of tools is a challenging task that has not been investigated deeply enough yet. Tool wear is a complex stochastic phenomenon [10]. In order to make forecasts of its evolution, it is necessary to define a significant indicator that represents it. Flank wear measurement is the most relevant and common degradation indicator for cutting tools [11,12]. Flank wear is quantified through the mean width of the wear land originated on the flank face of a cutting tool. Its computation follows international standards [13] and it will be indicated as *VB*.

Two main branches of research deal with the prognosis of cutting tool wear: namely, indirect and direct tool wear approaches [14]. Indirect methods consist in the evaluation of tool wear degradation through the use of cutting process quantities. The commonly adopted variables include axis torque or current [15,16], vibrations [17–25], acoustic emissions [19,21,23,25,26], spindle power [18,27–29] and cutting forces [10,11,19,21–25,30–36]. Main advantage of most indirect estimation techniques is that they could be developed to work in real-time, such that while the machine is running, an updated wear indicator is available. However, all indirect methods are limited by the fact that there are many other effects greater in magnitude than tool wear. This makes indirect observation method still immature for general cutting conditions [4], even if it is possible to find some research works that try to deal with the normalisation of indirect quantities with respect to cutting parameters [11,37,38]. On the other hand, direct measurement of flank wear is more robust against the cutting process variability. Direct inspection could be obtained through point wise scanning, profilometric acquisitions [39] or by 2D/3D calibrated picture analysis [39–41]. Direct methods still have some shortcomings. Firstly, uncertainties are related to the accuracy of the inspection system and the post-processing of the measurements. Secondly, in general machining processes should be stopped and the cutting tool exposed to the inspection system in order to measure the tool-tip status. Nevertheless, the inspection task could be performed at periodic stops in masked time, for instance at tool changes, or triggered by user defined events according to optimised production strategies.

When dealing with prognosis approaches, several researches tried to apply machine learning or data-driven methods (black-box models) for the prediction of tool wear. Black-box approaches try to learn complex relationships between indirect quantities and wear directly from data, and they are typically less interpretable [4]. Cheng et al. applied Support Vector Regression in order to predict flank wear evolution, starting from cutting forces, vibrational signals and machined surface pictures [42]. Guo et al. proposed a DenseNet algorithm in order to correlate flank wear to cutting forces, vibrations and acoustic emission signals. A multi-step encoder–decoder system predicted the short-term and long-term evolution of the flank wear [25]. A CNN was proposed by Zhang et al. to predict the *RUL* of cutting tools starting from current, vibrations and acoustic emission signals [9]. Yan et al. proposed ResNets, in order to predict flank wear evolution in TC4 titanium alloy milling based on Short-Time Fourier Transformation of vibrations and force signals. Nevertheless, future dataset expansion is needed to cope with other process parameters and processing environments [24]. Wang et al. conceived a hybrid methodology based on machine learning capable of integrating heterogeneous data (process parameters, power profiles and tool wear images). Wear severity was assessed by a convolutional neural network, while *RUL* predictions were performed by a recurrent neural network [28]. Model-based approaches (white-box models) define the degradation model on first principles and only few coefficients need to be tuned based on experiments [4]. Liu et al. proposed a cutting force model including tool wear information. The model-based approach could be used for tool flank wear predictions through cutting force and temperature measurements [36]. Statistical-based approaches (grey-box models), are based on user-selected dynamical models whose parameters are estimated through data [4]. Yu et al. applied a weighted HMM framework for the prediction of tool flank wear, based on vibrations and acoustic emission signals [21]. Zhang et al. proposed a particle filter scenario including a linear degradation model. Its parameters are updated online from in-process vibrational measurements [43]. In some cases, knowledge-based approaches (experience-based) are applied, where rule-based models are created upon expert knowledge [4], like in [32]. Scientific works were mapped in Table 1, according to several criteria, highlighting some useful aspects of the proposed methodologies like the availability of *RUL* PDF, the number of needed run-to-failures (RTFs), and algorithm domain.

Despite the amount of research works on tool prognostics, several challenges are common between them:

- need for several RTF experiments to train machine learning or data-driven models
- robustness of the solution with respect to cutting parameters
- prediction of the *RUL* PDF is missing
- algorithms are static, not updated through on-line measurements

The challenges above become more and more relevant when tool prognosis needs to be applied in one-of-a-kind or small batch production scenarios, where available data are limited and cutting conditions are varying, bringing to limited knowledge of the effect of parameters on tool lives. To overcome these limitations, a direct wear prognostic approach is here conceived. The solution is based on a hybrid adaptive algorithm, fusing the statistical framework with the data-driven world. In fact, a particle filter state observer is used to adapt on-line two different models: the weights of a simple multi-layer perceptron, and the coefficients of a cubic polynomial. These models map the flank wear degradation curve with respect to the insert cutting time. The algorithm is assumed to take as input flank wear measurements, typical of direct tool condition monitoring systems.

The structure of this paper is organised as follows: in Section 2, the experimental set-up and campaign are firstly described, then the conceived adaptive and hybrid prognosis methodology is formulated, together with the description of the used performance metrics. In Section 3, the analysis of the experimental data is discussed and the two versions of the hybrid methodology are compared. At last, conclusions are drawn in Section 4.

Table 1
Map of State-of-The-Art cutting tool prognosis approaches. Algorithms are described by application, employed sensors, measured features, class (i.e., direct, indirect or mixed — direct and indirect), algorithm and relative domain, number of experimental RTFs, *RUL* PDF availability.

Ref.	Machining process	Measurements	Features	Class	Algorithm	Domain	Varying cutting parameters	RTFs	*RUL* PDF
[31]	Broaching	Cutting forces, strain, vibration	PCA on force covariance matrix	Indirect	Support Vector Machine	Black-box	No	–	No
[15]	Drilling	Current	Energy of wavelet packet signal node	Indirect	ARMA	Grey-box	Yes	2	No
[18]	Turning	Flank Wear, power consumption, vibrations	*VB*, power mean, vibration moving average (separately)	Mixed	A sort of similarity method	Grey-box	No	30 (some of them discarded)	Yes
[20]	Milling	Vibrations	A set of time domain features	Indirect	Neuro-fuzzy Neural Networks	Black-box	No	7800 (points)	No
[17]	Milling	Cutting forces, vibrations, acoustic emissions	Root mean square (rms), standard deviation (std), peak (forces); rms, kurtosis (accelerometers); mean, std (acoustic emission)	Indirect	Dynamic Bayesian Networks	Grey-box & Black-box & experience-based	No	3	No
[32]	Milling	Cutting forces	Peak, amplitude, average, std	Indirect	Fuzzy inference	Experience-based	–	–	–
[30]	Milling	Cutting forces	Time-domain features	Indirect	Multiple regression models	Grey-box	No	2 (109600 points)	Yes
[33]	Milling	Cutting forces, vibration, acoustic emission	–	Indirect	Artificial Neural Network (Extreme Learning Machine)	Black-box	No	1 (945 points)	No
[19]	Milling	Cutting forces, vibration, acoustic emission	Time–frequency domain features	Indirect	Support Vector Regression	Black-box	No	2	Yes
[34]	Milling	Cutting forces	Energy from time–frequency domain features	Indirect	Continuous Hidden Markov Models and Gaussian Process Regression	Grey-box	No	7	No
[22]	Milling	Cutting forces, vibrations	Four time domain features	Indirect	Extreme learning machine	Black-box	No	1 (945 points)	Yes
[27]	Milling	Spindle power	Rms	Indirect	Artificial Neural Network	Black-box	Yes	4	No
[23]	Milling	Cutting force, vibration, acoustic emission (workpiece side)	Energy features from wavelet transform and blind source separation	Indirect	Non-linear regression with different model for each cutting tool tested	Grey-box	–	–	–
[35]	Milling	Cutting forces	Time and frequency domain features	Indirect	Bayesian multi-layer perceptron	Grey-box & black-box	Yes	12	Yes
[21]	Milling	Cutting forces, vibrations, acoustic emissions	rms (accelerometers)	Indirect	Weighted Hidden-Markov Models	Grey-box	No	3	No
[25]	Milling	Flank wear, cutting forces, vibrations, acoustic emission	Learned features (deep learning)	Mixed	DenseNets and auto-encoder	Black-box	No	2	No
[28]	Milling	Spindle power, milled surface images, tool wear images	Learned features (deep learning)	Mixed	Convolutional Neural Network and Recurrent Neural Network	Black-box	Yes	6300 images	No

(*continued on next page*)

Table 1 (*continued*).

Ref.	Machining process	Measurements	Features	Class	Algorithm	Domain	Varying cutting parameters	RTFs	*RUL* PDF
[24]	Milling	Vibrations and cutting forces	Short-Time Fourier Transformation	Indirect	ResNet	Black-box	No	7	No
[36]	Milling	Cutting forces and temperature	–	Indirect	Tool wear model estimation from cutting forces and temperature	Model-based	Yes	9	No
[10]	Micro-milling	Flank wear measurements, cutting forces	–	Mixed	Particle Filter and Long Short Term Memory network	Grey-box & black-box	Yes	9	Yes

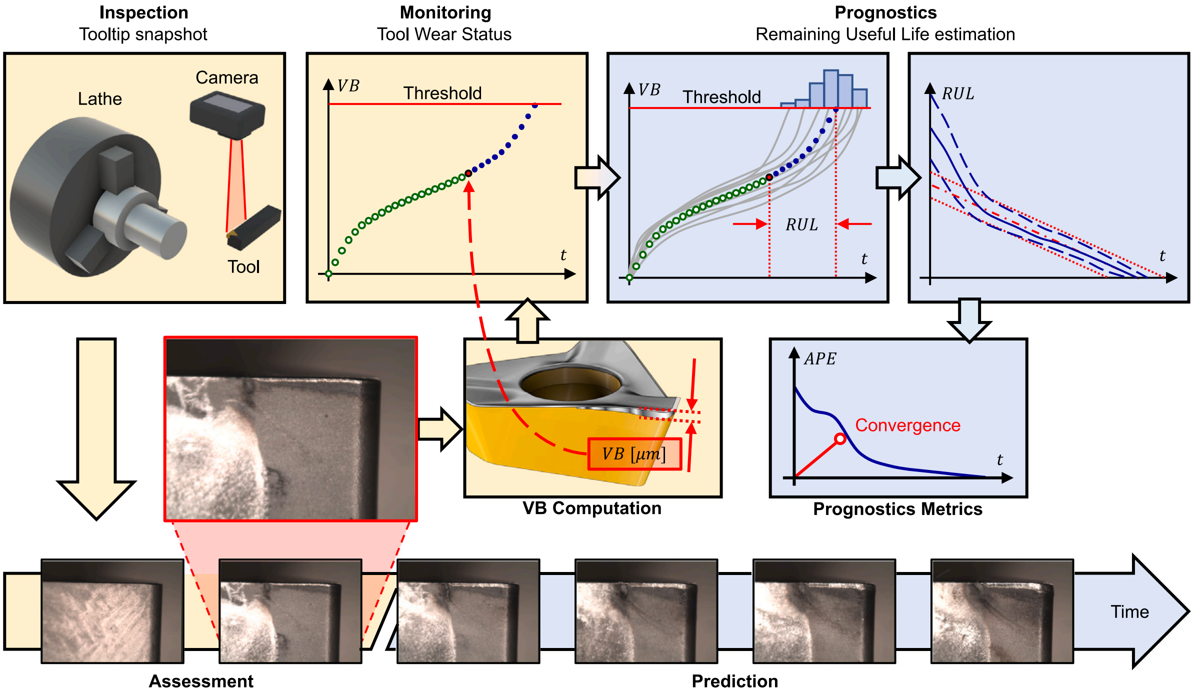


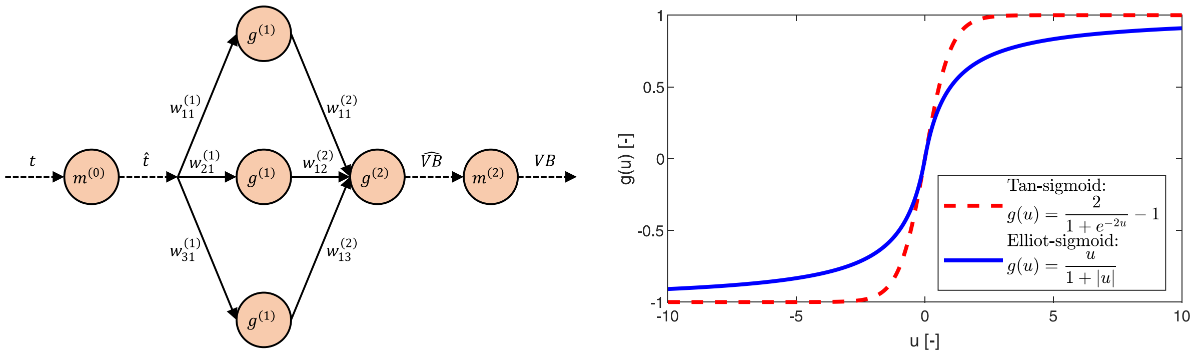
Fig. 1. Workflow for the conceived methodology. From tool pictures up to the estimation of the cutting tool Remaining Useful Life, as well as the prognostics metrics.

2. Materials and methods

In the next subsection, all the elements of the developed prognostics approach are presented. Following the general framework of Fig. 1, the solution is based on three steps: firstly a visual inspection of the cutting tool is carried out to obtain flank wear measurements; a hybrid particle filter framework, embedding multi-layer perceptrons or cubic polynomials, constitutes the monitoring and prognostics tool, allowing for the prediction of cutting tools' remaining useful life; at last, prognostics metrics are defined as means for prognosis performances and robustness analysis. The last subsection deals with the experimental set-up and campaign.

2.1. Particle filter hybrid framework definition

In this section, the hybrid (statistical-based and data-driven) adaptive approach for tool wear prognosis is presented. The solution is based on a combination of particle filter (PF) state observer, to set-up a bayesian update framework for a set of multi-layer perceptrons (MLPs) and a set of cubic polynomials (P3s), referred as data-driven models (DDMs), in general. The approach (in similar fashions) was investigated in other applications (such as crack growth prediction and lithium-ion batteries prognostics), highlighting its interesting adaptive capabilities [44,45]. Furthermore, this algorithm allows for the prediction of the cutting tool



(a) MLP.

(b) Activation function comparison.

Fig. 2. (a) Proposed MLP architecture. The input neuron normalises time values and propagates it to the hidden layer, which features three neurons. The normalised $\widehat{VB}$ is the result of the output layer, which is then mapped back to *VB*. (b) comparison between the tan-sigmoid activation function (red dashed line) [45] and the Elliot-sigmoid activation function (blue solid line) [46].

RUL PDF, required by the international standards [6] but rarely available on state-of-art approaches. In order to present the approach it is necessary to start with the description of the adopted DDMs.

2.1.1. Multi-layer perceptron

In the conceived scheme, the MLP represents a non-linear mapping of the flank wear width (*VB*, see Section 2.3) evolution with respect to insert cutting time t. MLP was chosen due to its demonstrated universal fitting capabilities [47], but other functions may be used (i.e., polynomials, see Section 2.1.2). In this case, the network complexity is reduced at minimum: the input and output layers are composed of one neuron; the hidden layer is composed of three neurons. The structure and nomenclature of the MLP is shown in Fig. 2(a). The input of the network (insert cutting time t) is normalised to $\hat{t} = m^{(0)}(t) = \hat{t}^- + (\hat{t}^+ - \hat{t}^-)(t - t^-)/(t^+ - t^-)$, i.e. mapped from the interval $[t^-; t^+]$ to $[\hat{t}^-; \hat{t}^+]$. The output $o_i^{(l)}$ of a neuron i in layer l is computed through Eq. (1):

$$o_i^{(l)} = g^{(l)}\left(\sum_{j=1}^{n_j} w_{ij}^{(l)} o_j^{(l-1)}\right) \tag{1}$$

where j indicates the neuron of the preceding layer $l-1$; $g^{(l)}(\cdot)$ is the activation function of layer l; $w_{ij}^{(l)}$ is the weight i of layer l, that multiplies the output $o_j^{(l-1)}$ of the previous layer $l-1$. In the conceived scheme two different activation functions are used, depending on the layer. In the hidden layer, Elliot-sigmoid function (Eq. (2)) was used [46]:

$$g^{(1)}(u') = \frac{u'}{1+|u'|} \tag{2}$$

where u' is a generic variable. A first difference with the architecture in [45] is found in the choice of the activation function. Cadini et al. adopted with success a tan-sigmoid activation function for crack growth prognosis, but the same function demonstrated not to be suitable for our scenario (tool prognosis). In fact, tan-sigmoid exhibits a fast transient towards the horizontal asymptote. The asymptote governs the curve fitting near the end-of-life of the tool. A fast convergence to the asymptote led in multiple cases to failures in finding the intersections with the prognosis threshold. The Elliot-sigmoid activation function is instead featured by a slower transient which solved the convergence problem (Fig. 2(b)). In the output layer, a pure linear (i.e., identity function) activation function was used, obtaining the normalised flank wear width $\widehat{VB}$. The resulting MLP non-linear function can be written as Eq. (3), where $VB_{MLP}(t)$ is the output and t is its input:

$$VB_{MLP}(t) = m^{(2)}\left(\sum_{j=1}^{3} w_{1j}^{(2)} \cdot g^{(1)}\left(w_{j1}^{(1)} \cdot m^{(0)}(t)\right)\right) \tag{3}$$

where the normalised output $\widehat{VB}$ is inversely mapped to $VB = m^{(2)}(\widehat{VB}) = VB^- + (VB^+ - VB^-)(\widehat{VB} - \widehat{VB}^-)/(\widehat{VB}^+ - \widehat{VB}^-)$, which restores the *VB* range from $[\widehat{VB}^-; \widehat{VB}^+]$ to $[VB^-; VB^+]$. Eq. (3) represents a non-linear parametric function of time, whose parameters are the weights of the MLP. Weights are initialised according to the Nguyen–Widrow method [48]. During the application of the algorithm, these parameters will be trained upon a single tool wear degradation curve. t^- and t^+ represent the minimum and maximum time values for the training RTF (thus $t^- = 0$), while VB^- and VB^+ are the minimum and maximum *VB* of the training RTF (thus $VB^- = 0$). $\hat{t}^-$ and $\hat{t}^+$, $\widehat{VB}^-$ and $\widehat{VB}^+$ are set by the user (here, they are 0 and 1, 0 and 1, respectively). The particular choice of these parameters, combined with the absence of bias terms in the MLP, allows the network to have a null intercept and to have only six parameters. The parameters, the MLP weights $w_{ij}^{(l)}$ are collected in the θ_{MLP} vector. These choices represent the second main difference with respect to the formulation in [45]. In [44], the authors adopted radial basis functions that led to a different formulation of the problem (compare Eq. (2) from [44] and (3) in this paper).

2.1.2. Cubic polynomials

An alternative scenario is represented by the utilisation of P3 to describe the tool wear degradation curve:

$$VB_{P3}(t) = at^3 + bt^2 + ct + d \tag{4}$$

In Eq. (4) the intercept (coefficient d) was set to zero, in order to generate a P3 passing through the origin. P3 curve was also designed to be a monotonically increasing function for non-negative values of t (negative values of t are not of interest, thus it is not necessary to over-constrain P3 coefficients in that domain). These assumptions led to the following sets of constraints for the P3 coefficients:

$$\begin{cases} a \geq 0 \\ b \geq 0 \\ c \geq 0 \end{cases} \qquad \vee \qquad \begin{cases} a > 0 \\ b < 0 \\ c > b^2/(3a) \end{cases} \tag{5}$$

As for MLP, P3 represents a non-linear parametric function of time, whose parameters are the coefficients a, b and c. The set of P3 coefficients is collected in the $\boldsymbol{\theta}_{P3}$ vector. During the application of the algorithm, these parameters will be trained upon a single tool wear degradation curve.

2.1.3. Particle filter

PF is a state observer capable to work with non-linear dynamical systems and non-gaussian states, disturbances and noises [4,49]. In the proposed prognostics scheme, the DDM describes a possible degradation curve of the tool (*VB* evolution over time). Although the degradation curve is initialised upon a historical RTF, it may not represent the actual degradation of the current cutting tool. Thus, the DDM parameters should be updated with online measurements of the cutting tool flank wear. To perform such operation, the DDM parameters, i.e., vectors $\boldsymbol{\theta}_{MLP}$ and $\boldsymbol{\theta}_{P3}$, are assumed to be states of a dynamical system, evolving during time according to the *process equation* (Eq. (6)):

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \boldsymbol{\omega}_k \tag{6}$$

where the index k represents the discrete time step; $\mathbf{x}_k$ is the state vector at time step k, containing the parameters of a DDM; $\boldsymbol{\omega}_k$ is a vector of random variables assumed to have variance proportional to the corresponding parameter $\boldsymbol{\omega}_k \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{k}})$; where the sth element of $\boldsymbol{\Sigma}_{\boldsymbol{k}}$ diagonal is $Q \cdot x_{k,s}$; Q represents disturbance intensity.

The *observation equation* associates the state vector (i.e., DDM parameters) to the system measurements, i.e. *VB* (Eq. (7)):

$$y_k = h(t_k, \mathbf{x}_k) + \eta_k \tag{7}$$

where y_k is the predicted *VB* through the DDM; t_k is the current cutting time; $h(t_k, \mathbf{x}_k)$ is either $VB_{MLP}(t_k)$ or $VB_{P3}(t_k)$, depending on the used DDM, with the parameters estimated at the current time step in the PF state $\mathbf{x}_k$; η_k is the measurement noise, following a normal distribution with null mean and standard deviation equal to R.

Objective of the PF is the description of the posterior PDF of the DDM parameters, conditioned on the observed data $p(\mathbf{x}_k|y_{1:k})$ [45]. The PDF of the DDM parameters is approximated by a set of N_p samples, referred as *particles*. In practice, N_p DDMs are generated, each of them creating a different map between insert cutting time and *VB*. A PF iteration consists of two phases. A prediction phase estimates the new particle values according to the process equation. At the end of this step a full set of new DDMs is obtained, which represents the "a priori" distribution of the DDM parameters (none of the measurements are used). An update phase improves the state PDF estimation. A likelihood equation $L(y_{1:k}|\mathbf{x}_k)$ is computed for each particle, accounting for all the available measurements from the beginning ($k = 1$) up to the discrete time step k (Eq. (8)):

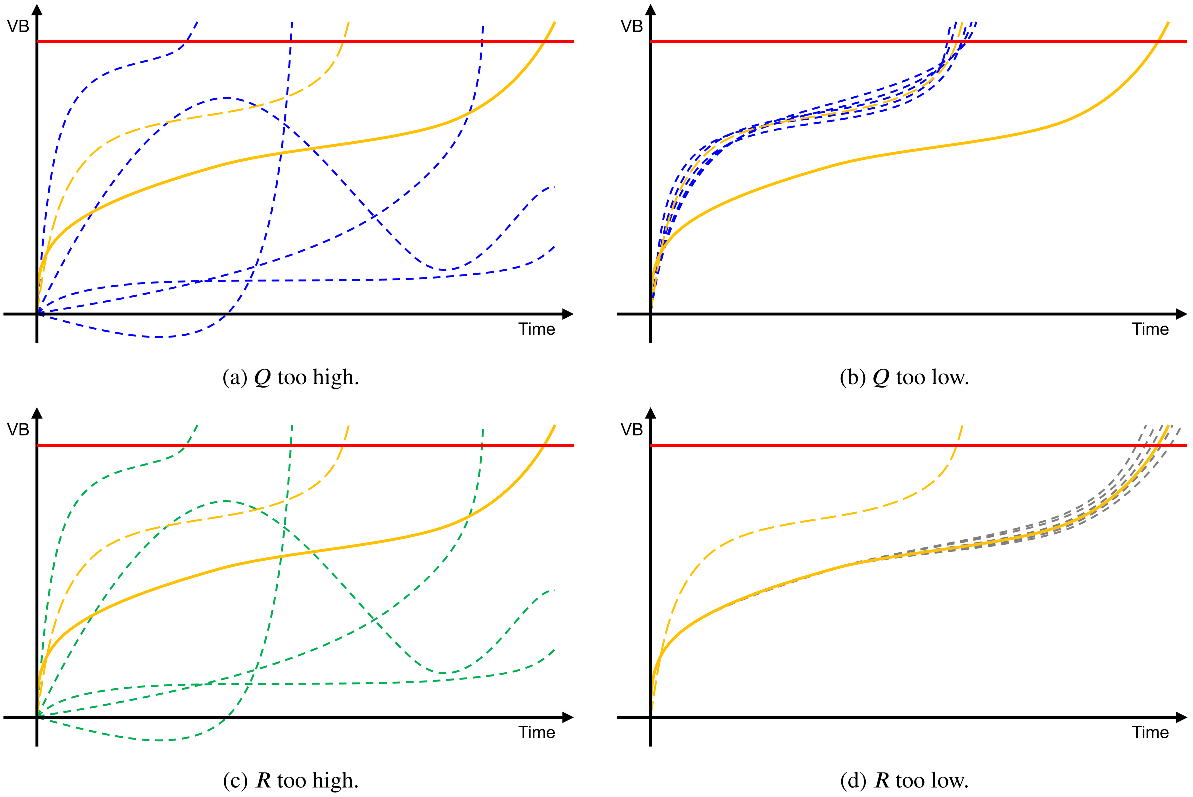
$$L(y_{1:k}|\mathbf{x}_k) = \exp\left(-\frac{1}{2R^2} \sum_{z=1}^{k} \left(y_z - h(t_z, \mathbf{x}_k)\right)^2 v_z(k) \right) \tag{8}$$

where z is the index used to represent each discrete time step up to k. A weight term, denoted as $v_z(k)$, is assigned to each time instant using the exponential law $v_z(k) = \exp(-\xi(k - z))$, driven by the non-negative parameter ξ (when ξ is set to 0, all samples are equally important; the larger the value of ξ, the quicker past samples lose their significance). Eq. (8) expresses the likelihood of observing the measured *VB* curve (giving more relevance to more recent measurements), depending on the current estimation of the state $\mathbf{x}_k$ described by a particle. Thus, the likelihood tells us how much we should trust that particle (i.e, the parameters of the associated DDM). Indeed, a weight $u_{k,s}$ is given to a particle s proportional to its likelihood (Eq. (9)):

$$u_{k,s} = \frac{L(y_{1:k}|\mathbf{x}_{k,s})}{\sum_{s=1}^{N_p} L(y_{1:k}|\mathbf{x}_{k,s})} \tag{9}$$

where $\mathbf{x}_{k,s}$ is the state represented by particle s. The PF iteration ends with the resampling stage (performed according to [45]), where the particles probabilities of being redrawn are proportional to their likelihood. The new set of particles represents the discrete approximation of the posterior PDF $p(\mathbf{x}_k|y_{1:k})$ of the DDM parameters. At this point the PF iteration is finished. The prediction and update stages are repeated every time a new flank wear measurement is available.

This implementation differs from [44,45], where exponentially decreasing variances were adopted. Despite their formulation allowed improving the *RUL* confidence intervals in time, the adaptability of the algorithm is reduced, and at the same time more hyperparameters have to be tuned. Here, the main hyperparameters are Q and R, whose effect is shown in Fig. 3:



(a) Q too high.

(b) Q too low.

(c) R too high.

(d) R too low.

Fig. 3. Effect of PF hyperparameters on the algorithm behaviour. Red solid horizontal line is the prognostics threshold; orange solid line is the validation degradation curve; orange dashed line is the training degradation curve; blue dotted lines are PF explorations of degradation curves; green dotted lines are resampled curves (accepted for the "a posteriori" *RUL* predictions); grey dotted lines are refused curves (not included in the "a posteriori" *RUL* predictions). In order to reach the desired algorithm performances, a trade-off between the four cases must be chosen. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

- a too high process disturbance (high Q) generates a completely random exploration of the DDM parameters' space. The DDM forgets its initialisation, which, in turn, provides random *RUL* predictions (Fig. 3(a)).
- a too low process disturbance (low Q) makes the algorithm not adaptive. The DDM parameters' space is not explored at all (Fig. 3(b)).
- a too high measurement noise (high R) means accepting all the "a priori" DDM fittings, even if they do not follow measurements (Fig. 3(c)).
- a too low measurement noise (low R) means to refuse a lot of "a priori" DDM fittings, leading to the preservation of a really limited number of DDM estimations, and losing any probabilistic description of the estimated quantities (Fig. 3(d)).

Q and R tuning is obtained through a trade-off between these limit cases.

2.1.4. Cutting tool RUL prediction

At each time instant t_k, the diagnosis phase can be performed after the update phase. Each particle provides the estimate for the cutting tool flank wear $VB_{k,s}$, by evaluating the associated DDM curve at t_k. Thus, the full set of particles provides the approximated posterior PDF of the flank wear $p(VB_k|y_{1:k})$. In order to realise the final objective of PHM, prognosis must be integrated. With this aim a threshold value VB_t for the limit flank wear was set to 150 μm. At every discrete time step k, after the update phase, it is possible to compute the estimated time of end-of-life $t_{EoL,k,s}$ of each particle s. This is performed by evaluating the associated DDM for future values of time, until the predicted *VB* intersects the threshold line. The time at which the DDM non-linear map crosses the threshold gives the $t_{EoL,k,s}$ estimate. The set of estimated end-of-life times, constitutes the approximated PDF $p(t_{EoL,k}|y_{1:k})$. *RUL* is computed as the difference between the time of end-of-life estimate $t_{EoL,k,s}$ and the prediction instant t_k, for every particle. As for the time of end-of-life, the set of *RUL* estimations gives the approximated posterior probability $p(rul_k|y_{1:k})$. We will denote the *RUL* prediction of the hybrid algorithm at time step k as the median of the "a posteriori" *RUL* PDF $RUL_k = med(p(rul_k|y_{1:k}))$. We also define the normalisation of *RUL* as $\rho_k = RUL_k/t_{EoL,true}$.

2.2. Prognostic metrics

Prognostics metrics are a valuable tool for measuring the performances of a prognosis approach. In addition to quantitative assessment they also offer a comprehensive visual perspective that can be used in designing the prognostic system. Furthermore, they allow comparing the performances of an algorithm with respect to others. Here, three main prognostics metrics [50] are used to analyse the behaviour of the conceived approach on the experimental tests and to compare the performances across the cases.

The first metric is the absolute prediction error (*APE*). *APE* is a function of the time step k, and it is computed as the absolute value of the difference between the true *RUL* at time step k ($RUL_{true,k}$) and the *RUL* prediction (RUL_k, Eq. (10)):

$$APE_k = |RUL_{true,k} - RUL_k| \tag{10}$$

where APE_k is the absolute prediction error at time instant t_k. *APE* furnishes a direct measure of the *RUL* prediction error during the cutting test. Here, it is used in place of relative error, since $RUL_{true,k}$ tends to zero as t_k approaches the time of end-of-life $t_{EoL,true}$ of the tool, leading by definition to an infinite relative error. Then, we define a normalised error $\psi_k = APE_k/t_{EoL,true}$. The second metric of interest is the prognostics horizon (*PH*). *PH* was proposed in two fashions [50]. Here, the probabilistic one is computed.[1] *PH* is defined by Eq. (11):

$$PH = \frac{t_{EoL,true} - t_{k_{\alpha\beta}}}{t_{EoL,true}} \tag{11}$$

where $t_{k_{\alpha\beta}}$ is defined as:

$$t_{k_{\alpha\beta}} = \min_k \int_{\alpha^-}^{\alpha^+} p(rul_k|y_{1:k})dt \geq \beta \tag{12}$$

where $\alpha^- = t_{EoL,true}(1-\alpha)$ and $\alpha^+ = t_{EoL,true}(1+\alpha)$, with α representing the allowable prediction error; β is the minimum admissible probability mass. Indeed, *PH* represents the percentage of tool life at which the required prognostics accuracy is met. As the *PH* increases, more time becomes available for intervention before tool failure.

The last prognosis metric is convergence (C). Convergence is a meta-metric quantifying the speed of decay of another prognosis metric [50]. Here, the convergence of the *APE* is proposed. It is defined by Eq. (13):

$$C_{APE} = \sqrt{APE_x^2 + APE_y^2} \tag{13}$$

where APE_x and APE_y are the coordinates of the centre of mass of the area under the APE curve on the time and APE axes, respectively. C_{APE} thus represents the euclidean distance from the origin of such centre of mass. The closer is this point with respect to the origin, the faster is the convergence of its metric (i.e., *APE*). Convergence is effective only for decreasing metrics [50]. Convergence is presented in its normalised fashion as: $\chi_\psi = \sqrt{\psi_x^2 + \psi_y^2}$ and it is computed on the $\lambda - \psi$ chart, where λ_k is the normalised cutting time: $\lambda_k = t_k/t_{EoL,true}$.

2.3. Experimental set-up and campaign

The experimental campaign consisted of a set of five RTFs in turning applications. The tests were performed on a SOMAB UNIMAB 400 lathe, equipped with an analogical numerical control. A carbide tool with a lead angle of 95° was adopted (ISO standard code: TNMG220404-M5 5625, tool radius equal to $r_\varepsilon = 0.4$ mm, rake angle of 13° and a relief angle equal to 0°, with $Al_2O_3 - TiCN$ coatings) and fixed on tool holder, ISO code MTJNL2525M22. Hardened and tempered steel bars of UNI $39NiCrMo3$ were used to perform the wear tests. RTFs were performed with conventional lubrication, using cutting fluid (oil–water emulsion with 5% of HOCUT 795 SC), adduced through nozzle on the cutting zone, in order to reproduce realistic industrial production scenarios.

Experimental tests were performed according to a full factorial design with one central point and no replicates (thus one test per corner and a central point test), following [51]. Two factors were chosen, namely cutting speed and feed per revolution. These two parameters were chosen since they have strong influences on the cutting tool lives [52]. The factor levels for the RTFs were reported in Table 2. The radial depth of cut was set to 2 mm. Direct measurements of tool flank wear were performed through recurrent visual inspection. Starting from calibrated macro pictures of cutting edges, the *VB* was computed with a manual procedure. However, this step does not limit the application of the conceived prognostic approach, since similar results can be obtained through machine vision algorithms, as in [39–41]. It is important to have high quality pictures of clean cutting edges and properly define a region of interest. Then, thresholding, contouring, aligning and measuring should be performed. A Stereomicroscope Optika SZN-T with Motic SMZ-168T support was used to take flank wear width (*VB*) images during the cutting tests. A RTF test was stopped when the measured *VB* overcame 0.15 mm. The corresponding time of end-of-life of the tool ($t_{EoL,true}$) was reported in Table 2. The threshold was chosen lower than what is suggested by ISO 3685 [13], in order to avoid too high dispersion of the results in terms of insert duration, seen in preliminary tests. The choice was taken to limit the experimental effort, too. In fact, carrying out many experiments up to a *VB* of 0.3 mm would have required too many machining tool hours, considering the fact that flank wear measurements take an amount of time comparable or higher than the effective machining time [26]. This would have lead to unsustainable experimental

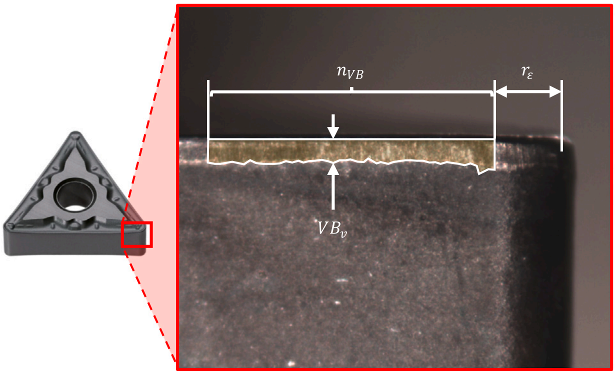


Fig. 4. Main quantities involved in the flank wear computation.

Table 2
Run-to-failure turning tests process parameters: cutting speed (v_c) and feed rate (c) are the two factors of the full-factorial design. Tool life, i.e. time of end-of-life ($t_{EoL,true}$), is also reported.

ID	v_c [m/min]	c [mm/turn]	$t_{EoL,true}$ [s]
1	220	0.1	765.6
2	220	0.3	345.6
3	190	0.2	798.2
4	160	0.1	2736.5
5	160	0.3	1062.1

Table 3
Chosen hyper-parameters for the conceived approach: N_p corresponds to the number of particles (equal for both cases); number of neurons per layer (only for MLP); Q is the process disturbances intensity (different for MLP and P3); R is the measurement noise standard deviation (different for MLP and P3); ξ is the hyperparameter for exponential weighting (equal for both cases); VB_t is the prognosis threshold (equal for both cases); α is the admissible prediction error (equal for both cases); β is the minimum acceptable probability mass (equal for both cases).

N_p [−]	Neurons [−]	Q_{MLP} [−]	Q_{P3} [−]	R_{MLP} [μm]	R_{P3} [μm]	ξ [−]	VB_t [μm]	α [−]	β [−]
5000	1-3-1	0.003	0.08	2.5	1.5	0.1	150	0.2	0.95

costs. Optimal cutting speeds and feed rates were adopted under conventional lubrication of steel cutting. The computation of the *VB* is performed following international standards [13] (Fig. 4).

As can be seen from Fig. 4, the flank wear width determination starts from a set of n_{VB} local measurements VB_v. The region of interest starts at a distance from the edge equal to the tool radius. Each measurement accounts for a small finite width Δl_v. Thus, the average flank wear *VB* can be computed over the region of interest by Eq. (14):

$$VB = \frac{1}{n_{VB}} \sum_{v=1}^{n_{VB}} VB_v \Delta l_v \tag{14}$$

3. Results and discussion

Results are presented in two phases. In a first section (Section 3.1), the adaptability of the conceived approach, *RUL* estimation and metrics will be discussed for a single train-validation RTFs combination. The objective of this first section is to provide an explanatory example, to show and describe the useful features of the algorithm and the related metrics. The reference case is presented for MLP only, and more specifically it represents the case when the MLP is initialised on RTF 5 and the algorithm is applied for the prognosis of the cutting tool of RTF 1. A second section (Section 3.2) deals with the discussion of the whole set of results, thus all the combination of train and validation RTFs. This section is divided in MLP and P3 results. A general analysis of the algorithm performances on all the RTFs is thus provided in this section. Here, the algorithm will be validated with a particular approach, inspired by Leave-One-Out Cross-Validation (LOO-CV), which will validate the hybrid framework performances right after the first DDM training RTF is available. This strategy is adopted to evaluate the performances on the first algorithm run. In both the cases, the hyper-parameters selected for the algorithms are reported in Table 3.

[1] This version of *PH* can be adopted only for prognosis algorithms providing PDF estimates of the *RUL*. A deterministic version of *PH* is described in [50] for algorithms providing only *RUL* point estimates, too.

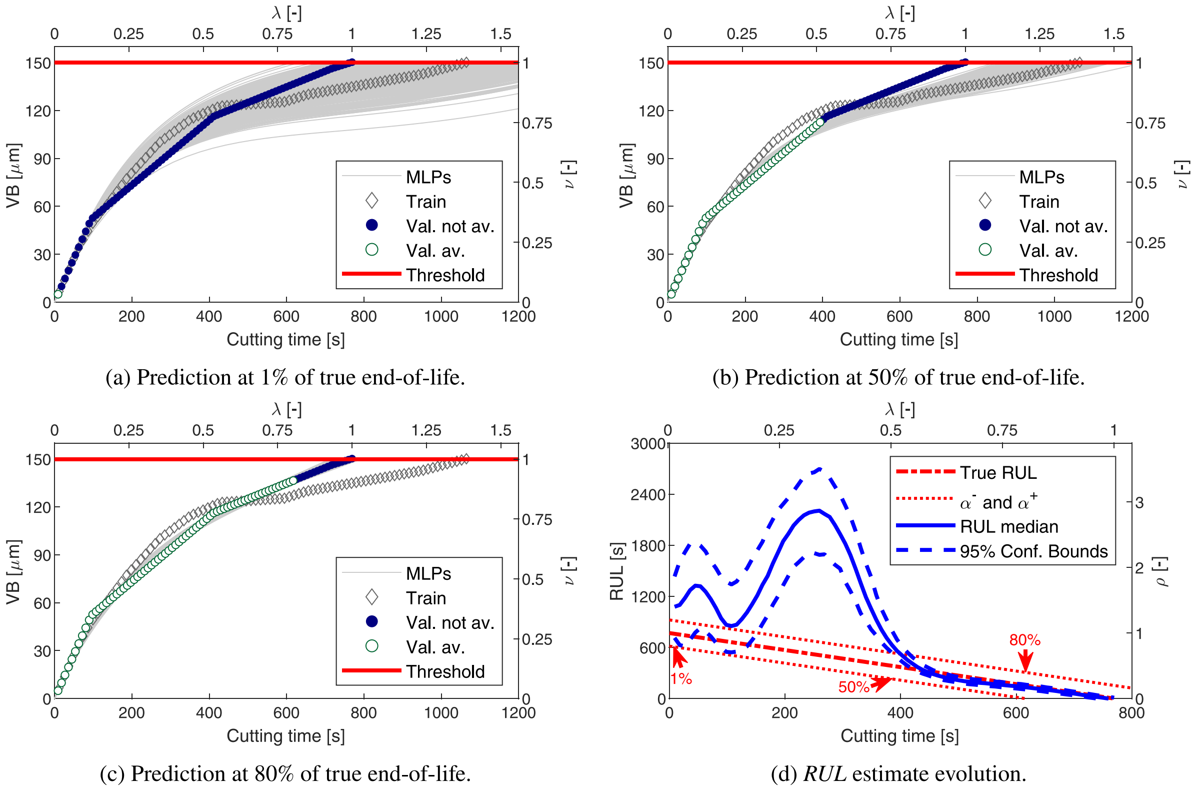
3.1. Single train-validation combination

In this section, the performances of the algorithm and the prognostics metrics are analysed for a reference combination of train-validation RTFs. More specifically, training is performed on RTF 5 and validation on RTF 1. Results are reported for the hybrid PF+MLP algorithm. This particular combination, allows to clearly identify the adaptability characteristic of the conceived approach and to discuss the adopted metrics. Given that certain performance metrics and *RUL* are inherently affected by the end-of-life of the analysed tool, we opt for generalisation and effectiveness in result comparison by presenting their normalised values. Thus, time-evolving results are plotted as functions of λ. *VB* axis is normalised as $v_k = VB_k/VB_t$, *RUL* axis is normalised as ρ and *APE* axis is normalised as ψ. The discussion of this section makes reference to Fig. 5, and reports also the relationship between each quantity and its normalised counterpart. MLPs' configurations and evolution along the RTF, as a consequence of the PF update, are presented at three different instants, i.e. 1% (Fig. 5(a)), 50% (Fig. 5(b)) and 80% (Fig. 5(c)) of the validation cutting tool life. This choice was taken for conciseness reasons, while keeping enough information content. First of all, in Fig. 5(a), it is possible to see the difference between the training and validation flank wear evolutions (diamonds and circular data points, respectively). Such difference is mainly associated to the cutting parameters adopted for the RTF and secondly, to the stochastic nature of the tool wear phenomenon: training RTF 5 was performed at a low cutting speed and high feed rate, whereas validation RTF 1 was performed at high cutting speed and low feed rate. As expected, during RTF 1, the cutting tool degradation was faster with respect to RTF 5. At the top of the graph, a solid red line represents the selected prognosis threshold, thus the flank wear limit for *RUL* prognosis. Diamond grey points were used to initially train the MLP. Thus, in a practical scenario, diamond data points constitute the only historical RTF needed to train and run the approach. Green circles are the data points received by PF from the field up to the current prediction time. As long as the RTF goes on, more experimental data (blue points) become available. Grey lines show the MLP curves associated with the particle distributions. 1% of cutting tool life, represents a condition in which the algorithm is still very confident about the initialisation phase. Thus, the set of MLPs faithfully pursuits the training RTF data. Nevertheless, the process equation disturbances are already trying to further explore the MLP fitting space. Since the prediction is made extremely early in the tool life, the spread of the MLP curves at the threshold of flank wear is quite wide. Intersecting the MLP curves with a vertical line at the prediction time (corresponding to the last green circle) and illustrating the corresponding values in a histogram would offer a graphical representation of the approximate posterior PDF of flank wear $p(VB_k|y_{1:k})$. Another intersection can be performed between MLP curves and the threshold line, providing a visual representation of the estimated *RUL* PDF at the current prediction time $p(rul_k|y_{1:k})$. In Fig. 5(d), the main quantities related to $p(rul_k|y_{1:k})$ are drawn as a function of time: namely, *RUL* estimate median (blue solid line), 2.5% and 97.5% percentiles, representing the 95% confidence bounds of the predictions (dashed blue lines). Their evolution is compared to the ground truth (true *RUL* red dash–dot line) and the prediction acceptability bounds (red dotted lines). As time passes by, the algorithm makes use of more experimental data to adapt to the current tool degradation rate. At 50% of tool life (Fig. 5(b)), the MLP curves lay on the available validation experimental data, causing MLPs *RUL* predictions to reduce their spread and shift towards the true RUL. Two regions should be emphasised in the comparison between training and validation *VB* curves, to better understand the algorithm behaviour: training *VB* degrades more rapidly in the region with λ values between about 0.25 and 0.35; training *VB* exhibits a flatter trend beyond 0.35. In the first region the algorithm starts to adapt the initial part of the MLP curves causing the *RUL* predictions to shift further away from the true *RUL* (note the maximum of *RUL* prediction in Fig. 5(d), where the evolving behaviour of the algorithm is summarised) and continuously overestimate it. At the beginning of the second region the PF effectively acts on the flatter part of the MLP curve by increasing its slope, and the predictions starts to converge towards the true *RUL*. Thus, at 50% of validation tool life (Fig. 5(b)), the algorithm enters into a more stable region, where the *RUL* median become more reliable and more accurate. It is worth noting that the algorithm progressively gains confidence in its predictions, resulting in MLP curves that more closely align with the experimental data and exhibit more uniform shapes; as a consequence, the prediction intervals for *RUL* are narrowing, too. Near the end-of-life of the cutting tool 5(c), the algorithm is fully adapted to the validation tool wear trend and the algorithm has finally converged to the true *RUL*.

As previously explained, prognostic metrics allow quantifying the performances of the algorithm, comparing different algorithms and different scenarios. Here, the metrics are computed with respect to the reference train-validation combination. The first metric (APE) is reported in Fig. 6. *APE* gives an immediate visual idea on how the algorithm is converging to the ground truth *RUL*. The same phases described above can be identified here. At the beginning of the RTF, the algorithm uses the information coming from the training data. The initial *APE* is approximately equal to the difference between validation and training time of end-of-lives. Within 25% and 35% of cutting tool life the algorithm adapts the first region of the MLP curves, leading to an overestimation of the *RUL*. In the second region between $\lambda = 0.35$ and $\lambda = 0.5$, the algorithm starts adapting the flatter part of the MLP curves; as a consequence the algorithm starts converging towards the true *RUL* of the validation cutting tool. Beyond $\lambda = 0.5$ the error becomes negligible and the algorithm predicts in advance the end-of-life of the tools.

The convergence metric is shown in the *APE* chart, too. As seen in Section 2.2, its value can be computed as the distance of the red circle (the centre of mass of the area beneath the *APE* curve) from the origin of the chart. Its normalised counterpart is computed on the $\lambda - \psi$ chart (top and right axis of Fig. 6). The normalised convergence for this RTFs combination is 0.79. This value provides a benchmark for the convergence time of the algorithm in case other algorithms are tested on the same scenario. In the next section, convergence will be discussed much further in order to compare different train-validation scenarios.

The prognosis horizon is partially represented in Fig. 5(d). In fact, the two red dotted lines represent the allowable prediction bounds, determined by the α value. *PH* is defined as the percentage of cutting tool life during which the beta criterion of Eq. (12) is satisfied. In this case, the criterion is satisfied from $\lambda = 0.55$ up to the cutting tool end-of-life. *PH* is thus equal to 45% of cutting tool life. This means that when the 55% of cutting tool life is expired, the algorithm accurately and robustly predicts the *RUL*. As *PH* increases, more time becomes available for preparing maintenance actions.



(a) Prediction at 1% of true end-of-life.

(b) Prediction at 50% of true end-of-life.

(c) Prediction at 80% of true end-of-life.

(d) *RUL* estimate evolution.

Fig. 5. *RUL* predictions when training on RTF 5, and validating on RTF 1. (a), (b) and (c) figures show the adaptability of the prognosis approach. Grey diamonds represent the training data for MLP initialisation. Green circles are the already available flank measurements. Blue points are the upcoming experimental points. Red line is the end-of-life threshold. Grey lines are the MLPs generated by the resampled particles. Blue solid line in (d) represents the evolution of the estimated *RUL* (i.e., *RUL* median); blue dashed lines represent the associated 95% confidence bounds; red dash–dot lines is the true *RUL*. Red dotted lines describe the allowable prediction bounds. Red arrows indicate the position of the (a), (b) and (c) figures in the full *RUL* prediction history. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

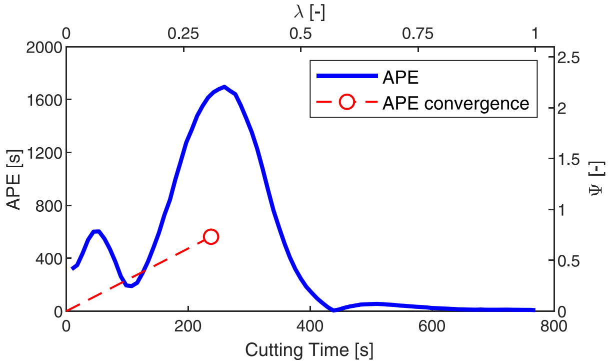


Fig. 6. Absolute prediction error APE (blue solid line) as a function of the normalised cutting time λ. Its convergence is represented with the red circle and red dashed line. This is the case in which training was performed on RTF 5 and validation on RTF 1.

3.2. Full set performances

In this section, a cross-validation inspired approach is presented. Since the objective of the conceived algorithm is to reduce as much as possible the experimental effort to perform prognostics, the validation of the devised method is carried out on a grid of cases. The grid is composed by different combinations of train and validation RTFs. Thus, at each iteration, one of the RTFs in Table 2 is used for DDM initialisation, while the others are used to discuss the adaptability performances of the hybrid methodology. In this sense, the proposed validation approach differs from LOO-CV, since LOO-CV would consist of iteratively training the algorithm on all the RTFs but one (used for validation). Here, the idea is to analyse how the algorithm works with different training conditions

Table 4
Prognostic metrics quantitative results for PF+MLP.

(a) Prognostics horizon.

PF+MLP	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5
Train 1	99%	52%	41%	32%	34%
Train 2	66%	99%	53%	0%	42%
Train 3	99%	55%	37%	0%	24%
Train 4	30%	56%	4%	37%	0%
Train 5	45%	39%	10%	22%	23%

(b) Normalised convergence.

PF+MLP	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5
Train 1	0.40	0.48	0.49	0.43	0.41
Train 2	0.33	0.52	0.35	0.66	0.37
Train 3	0.50	0.42	0.50	0.57	0.45
Train 4	6.65	1.91	1.49	0.37	8.03
Train 5	0.79	0.42	0.55	0.44	0.45

and also suggest the best cutting parameters choice for the training RTF. With this in mind, the reported results refer to the minimum performances obtainable with the algorithm. Improvements may be obtained by training the DDM on more tests.

3.2.1. MLP results

Two figures are reported representing this scenario. The first one reports the *RUL* predictions, the bounds for *PH* computation, true *RUL* and flank wear data for the training and validation RTFs (Fig. 7). In this sense, this figure is equivalent to Fig. 5(d) but reported for the whole grid of combinations. The second one is the summary of prognostics metrics for the whole set of combinations (Fig. 8). In both graphs the training RTF number corresponds to the row index, while the validation one to the column index. For clarity, graphs will be referred by the letter indicated in the box.

The first results to be observed, are the ones on the main diagonal of Fig. 7. These represent the cases where the train RTF is the same as the validation one. In Fig. 7A-G-M-S it can be seen that, as expected, the *RUL* predictions are accurate and the confidence bounds are almost completely enclosed in the allowable prediction bounds for a reasonable part of the tool life. This is confirmed by Table 4, where the *PH*s of all the diagonal terms are large (99%, 99%, 37% and 37%, respectively). This means that *RUL* is computed correctly from the beginning in A and G cases, and when 63% of tool lives are passed for cases M and S. *APE* plots in Fig. 8A-G-M-S and the convergence table (Table 4) are in agreement. In fact, these diagonal elements exhibit low values for both the metrics when compared to other grid cases. The grid element (5–5) (Fig. 7Y) exhibits a behaviour similar to that of case S; however, the *RUL* starts to consistently fall within the acceptability bounds at $\lambda = 0.77$. *PH* highlights this behaviour by reaching a value of 23%, showing a convergence in the final part of cutting tool life of case Y. Indeed, all the combinations on the diagonal exhibit *PH*s greater than 20%. Cases facing a *PH* greater than 20% are highlighted in Figs. 7 and 8 by axes coloured in green; cases that do not meet such criterion are represented with axes in red.

The second possible analysis regards the identification of critical tests and the associated cutting parameters. In order to perform such analysis it is necessary to look at a column per time. In fact, a column represents all the training scenarios for which the validation RTF was the same. By looking at Fig. 7, two columns seem more critical than others and face some issues in the adaptability of the prognosis approach: namely, column 3 and column 4. The fourth column is well estimated when training is performed on RTFs 1-4-5 (Fig. 7D-S-X). Predictions are instead unreliable for cases I and *N* (Fig. 7I-N). This behaviour is evident even from the metrics perspectives (Fig. 8), where *APE* is constant during the whole lives of cases I and N. The algorithm is thus facing trouble in the adaptation with these RTFs combinations. Column 4 of Table 4 provides similar information: on average, it exhibits the lowest *PH*s of the grid; rows 2 and 3 have null PHs. Convergence (Table 4) reaches higher values for cases I and *N* (row 2 and 3) than for the other combinations in the column, pointing out the worse performances of the algorithm in these cases.

As for column 4, 2 cases out of 5 in column 3 have a *PH* below 20%. In three training cases (Fig. 7C-H-M) the algorithm correctly converge to the true *RUL* trajectory. The algorithm results are stable and reliable for a large portion of the tool life, such that these cases have *PH*s of 41%, 53% and 37%, respectively. The same can be seen from the *APE* graphs in Fig. 8, where the prediction errors remain really low in most part of the tool lives. Convergence of the *APE* exhibits low values for these cases. Cases W and R are instead more critical for the convergence and stability of the algorithm. In fact, there is a small region near the end-of-life of the tool (λ comprised between 0.7 and 0.9) where the predictions of the algorithm overestimate the true *RUL* of the tool. This behaviour is in general associated to fast and drastic changes in the flank wear evolution of a cutting tool. This phenomenon is visible in Fig. 7R, where the test *VB* features constant degradation rate for the main part of tool life (up to $\lambda = 0.75$), when a sudden change in the rate is found. For the main part of the tool life, the training and test curves almost overlap, telling the algorithm that the current MLP curves are really good predictors and only small adaptations are required. Despite this, the last parts of MLP curves (the ones responsible for *RUL* predictions) are dramatically overestimating tool life. Nevertheless, these data points are not available to the algorithm until the end-of-life is almost reached. As soon as they have been acquired, the PF tries to follow the new degradation rate and adapts really fast to them. Convergence is particularly effective in highlighting this behaviour of the algorithm (see column 3, row 4 of Table 4). A similar behaviour is the cause of failure in the prediction for the case T, and for the sudden increase of *RUL*

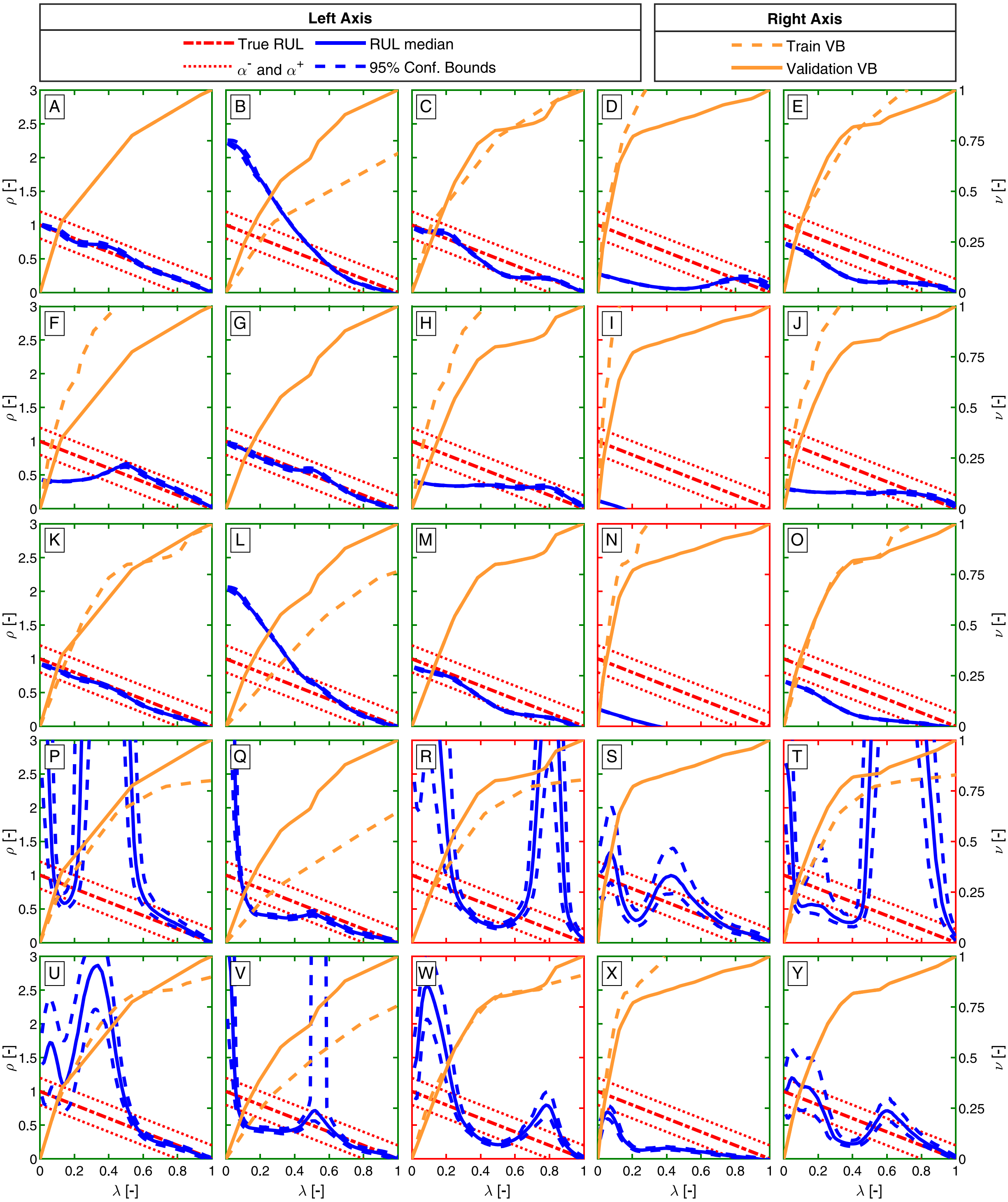


Fig. 7. Row index corresponds to the train RTF number, whereas column index to the validation RTF number. Graphs are identified by the letter in the box. *RUL* predictions (blue solid lines) and their confidence bounds (blue dashed lines) are compared to the true *RUL* (red dash–dot line) and the allowed prediction bounds (red dotted lines). Training (orange dashed line) and test (orange solid line) flank wear measurements are also compared. The graphs are reported on the normalised axes. Axes are coloured in green for cases with $PH \geq 20\%$, in red for cases with $PH < 20\%$. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

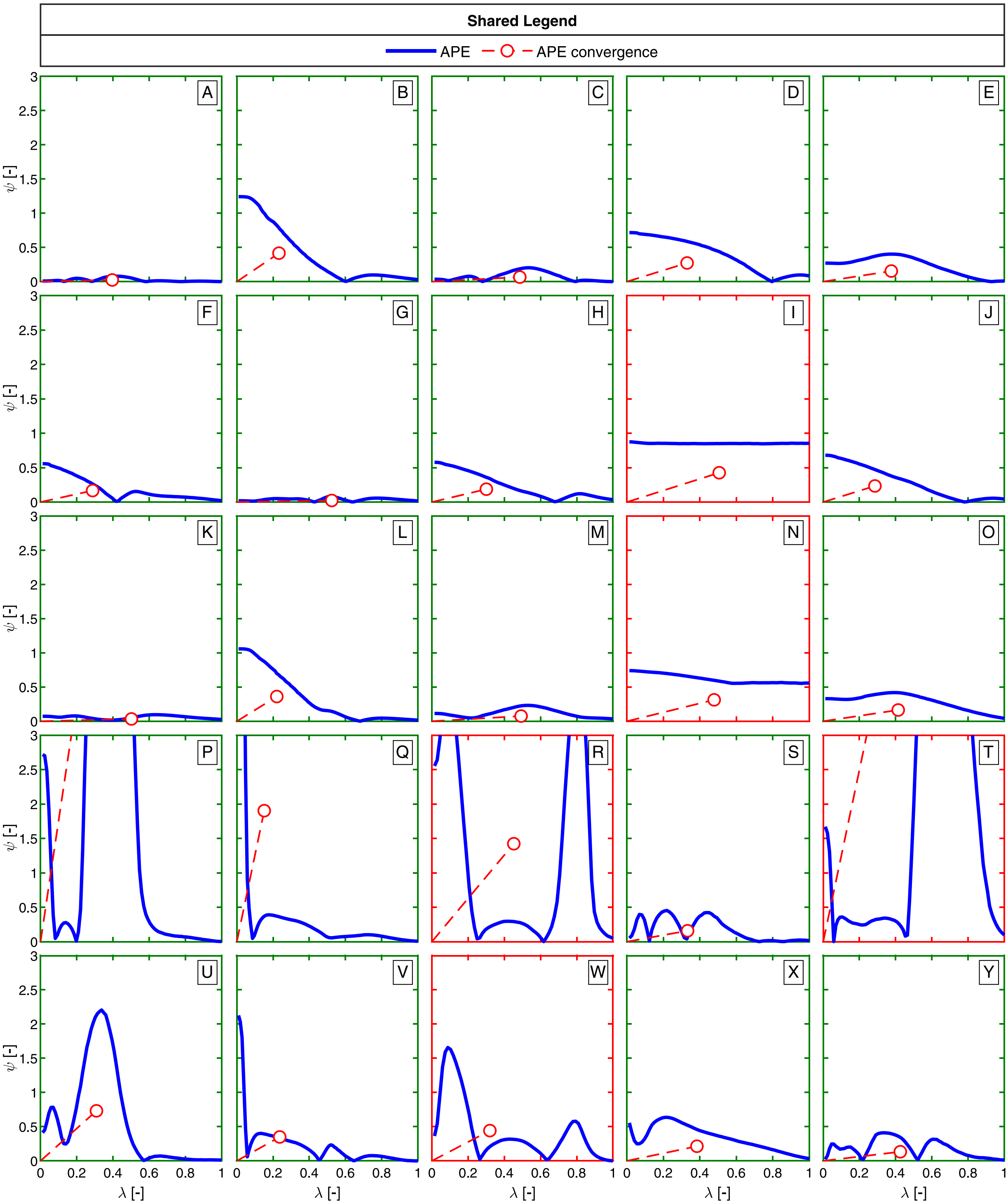


Fig. 8. Row index corresponds to the train RTF number, whereas column index to the validation RTF number. Graphs are identified by the letter in the box. Prognostic metrics (APE and C_{APE}) are represented for each train-validation combination on normalised axes. Axes are coloured in green for cases with $PH \geq 20\%$, in red for cases with $PH < 20\%$. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 5
Prognostic metrics quantitative results for PF+P3.

(a) Prognostics horizon.

PF+P3	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5
Train 1	34%	27%	21%	22%	29%
Train 2	33%	27%	21%	23%	28%
Train 3	34%	28%	21%	23%	28%
Train 4	14%	30%	16%	22%	29%
Train 5	33%	27%	21%	23%	28%

(b) Normalised convergence.

PF+P3	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5
Train 1	0.37	0.41	0.40	0.43	0.37
Train 2	0.40	0.38	0.43	0.44	0.38
Train 3	0.37	0.40	0.41	0.44	0.38
Train 4	0.42	0.44	0.46	0.44	0.42
Train 5	0.38	0.40	0.41	0.43	0.37

prediction in case P, in correspondence of λ between 0.2 and 0.6. Even for these cases the normalised convergence metric exhibits really high values (8.03 and 6.65, respectively), compared to the other cases in the grid. In general, the results suggest the fact that longer RTFs (low cutting speed) are more critical to adapt to.

The last analysis to be performed is the selection of the best training experiment. From a practical perspective, this analysis helps a company to decide how to select the parameters for the training RTF(s) to be performed. To this objective, it is necessary to have a look at Figs. 7 and 8, and Table 4 row-wise. In fact, once the row is fixed, the corresponding training RTF is set and the performances are validated on all the other RTFs. Referring to Fig. 7, all the predictions in row 1 converge to the *RUL* ground truths of the validation tests. The *RUL* median is smooth and stable and the algorithm shows good confidence. In Table 4, RTF 1 is the only training RTF with showing good predictions for all the cases, i.e., all the *PH*s are greater than 20%. Convergence shows stable results and low values (below 0.49), confirming that this choice of parameters (high cutting speed, low feed rate) is by far the best option for training the algorithm. Rows 2, 3 and 5 are fair options for training, too. 80% of the cases show *PH*s higher than 20%. In general when training on faster RTFs (high cutting speeds) seem to provide more stability, accuracy and robustness to the algorithm. Convergence metric is in accordance, too. Case 4 is the worst scenario, not allowing for a good PH for RTFs 3 and 5 (cases R and T). Nevertheless, although the predictions for this scenario are not the best, 60% of the cases have a *PH* greater than 20%. This means that the algorithm performances are, in general, reliable despite the choice of the RTF cutting parameters.

3.2.2. P3 results

The results relative to *RUL* predictions for the hybrid implementation of P3 are reported in Fig. 9. As for the MLP case, this figure is equivalent to Fig. 5(d) but reported for the whole grid of combinations. A second figure (Fig. 10) summarises prognostics metrics for the whole set of combinations. In both graphs the training RTF number corresponds to the row index, while the validation one to the column index. For clarity, graphs will be referred by the letter indicated in the box.

First and foremost, this hybrid solution allows *PH*s to reach the 20% threshold on 92% of the cases. This implies that in nearly all cases, the *RUL* converges to the true *RUL* with a reasonable lead time, leaving room to the operator to change the tool in time. Having a look at *RUL* predictions in Fig. 9 and errors in Fig. 10, the adaptation of the predictions shows common features among the validation cases. Convergence table highlights a similar behaviour of the algorithm throughout the tests, with uniform values comprised between 0.37 and 0.46. In the next paragraphs the results will be commented as for the PF+MLP case, trying to highlight the differences between the reported cases.

The first considerations to be drawn are those related to the main diagonal of Fig. 9. These represent the cases where the train RTF is the same as the validation one. In Fig. 9A-G-M-Y it can be seen that, as expected from the *PH*s, the *RUL* predictions are accurate for more than the 20% of the tool life. In particular, cases A-G-Y show the highest *PH*s equal to 34%, 27% and 28%, respectively. Cases M-S feature lower *PH*s: 21% and 22%. A first comment should be done on cases A-G-M and Y, since *RUL* predictions of these combinations feature similar behaviours (actually the shape of the curve is common for most of the combinations of the grid, with slight differences). The prediction of PF+P3 exhibits similar behaviour in these tests: a first drop in the *RUL* is present at the beginning of the tool life, that leads to a constant underestimation of the tool life; a fast adaptation of the polynomials generates a second region in which the *RUL* prediction is stable; at last, a final convergence brings the *RUL* prediction towards the real *RUL* near the end-of-life of the tool. *APE* plots in Fig. 10A-G-M and Y are in agreement and show the same three regions described for the *RUL* predictions. In Table 5, cases A-G-Y three diagonal elements feature the lowest convergence values, highlighting the fastest adaptation of the *RUL* predictions, while for case M, convergence assumes a higher value, 0.41, in accordance to the lower PH. Comments on case S will be done together with column 4 of cases in the next paragraph.

The identification of critical tests and the associated cutting parameters is performed by looking at the grid column-wise. By looking at Figs. 9, 10 and Table 5, columns 1 and 3 are the only ones showing a *PH* below the 20% of the tool-life. Anyway the performances on the rest of the combinations are good and satisfactory. An extended comment should be dedicated to column 4 (the axes of charts associated to these tests in Figs. 9 and 10 are highlighted in magenta). For these cases, the *RUL* predictions are

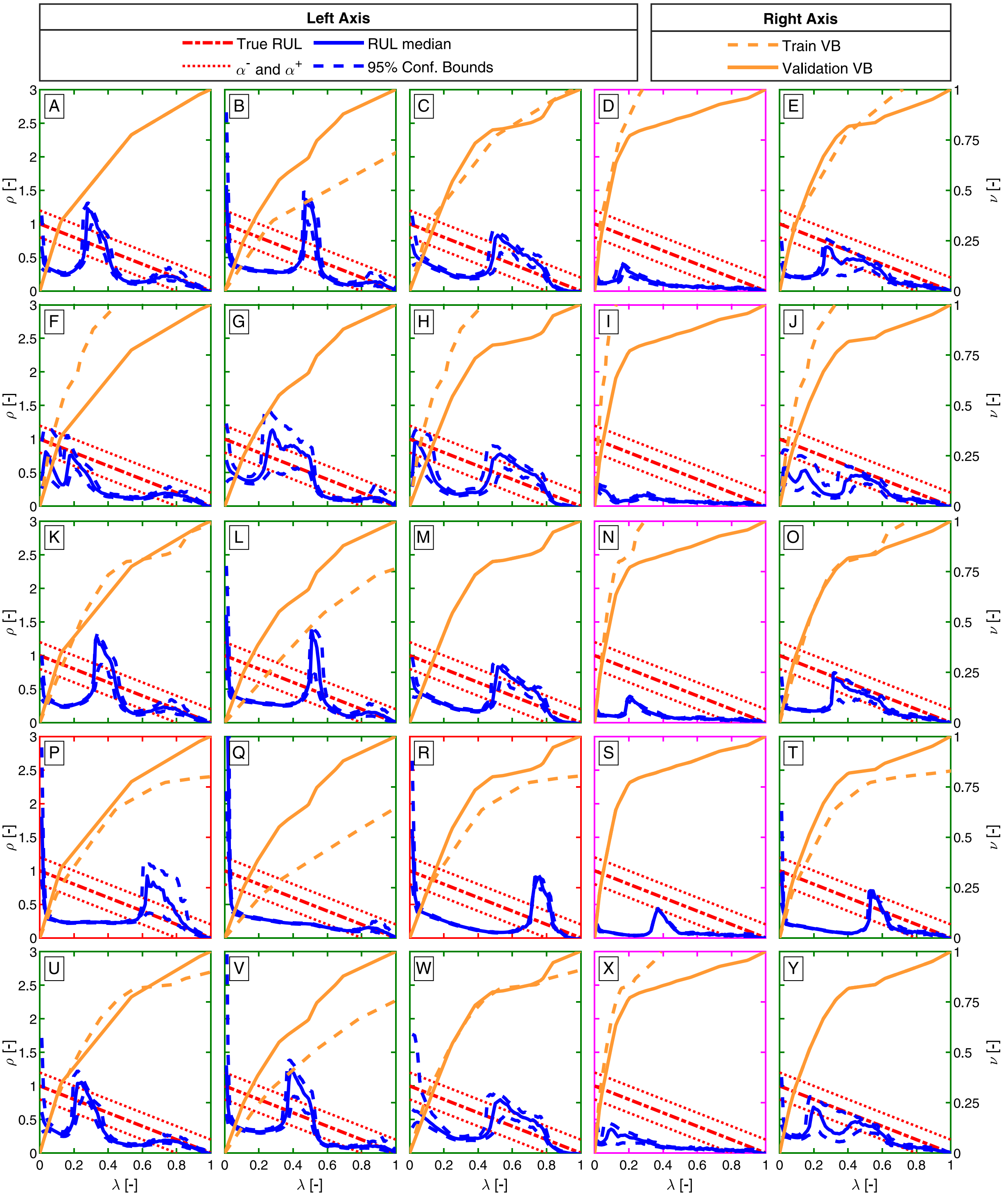


Fig. 9. Row index corresponds to the train RTF number, whereas column index to the validation RTF number. Graphs are identified by the letter in the box. *RUL* predictions (blue solid lines) and their confidence bounds (blue dashed lines) are compared to the true *RUL* (red dash–dot line) and the allowed prediction bounds (red dotted lines). Training (orange dashed line) and validation (orange solid line) flank wear measurements are also compared. Axes are coloured in green for cases with $PH \geq 20\%$, in magenta for cases with $PH \geq 20\%$ but critical for a practical implementation, in red for cases with $PH < 20\%$. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

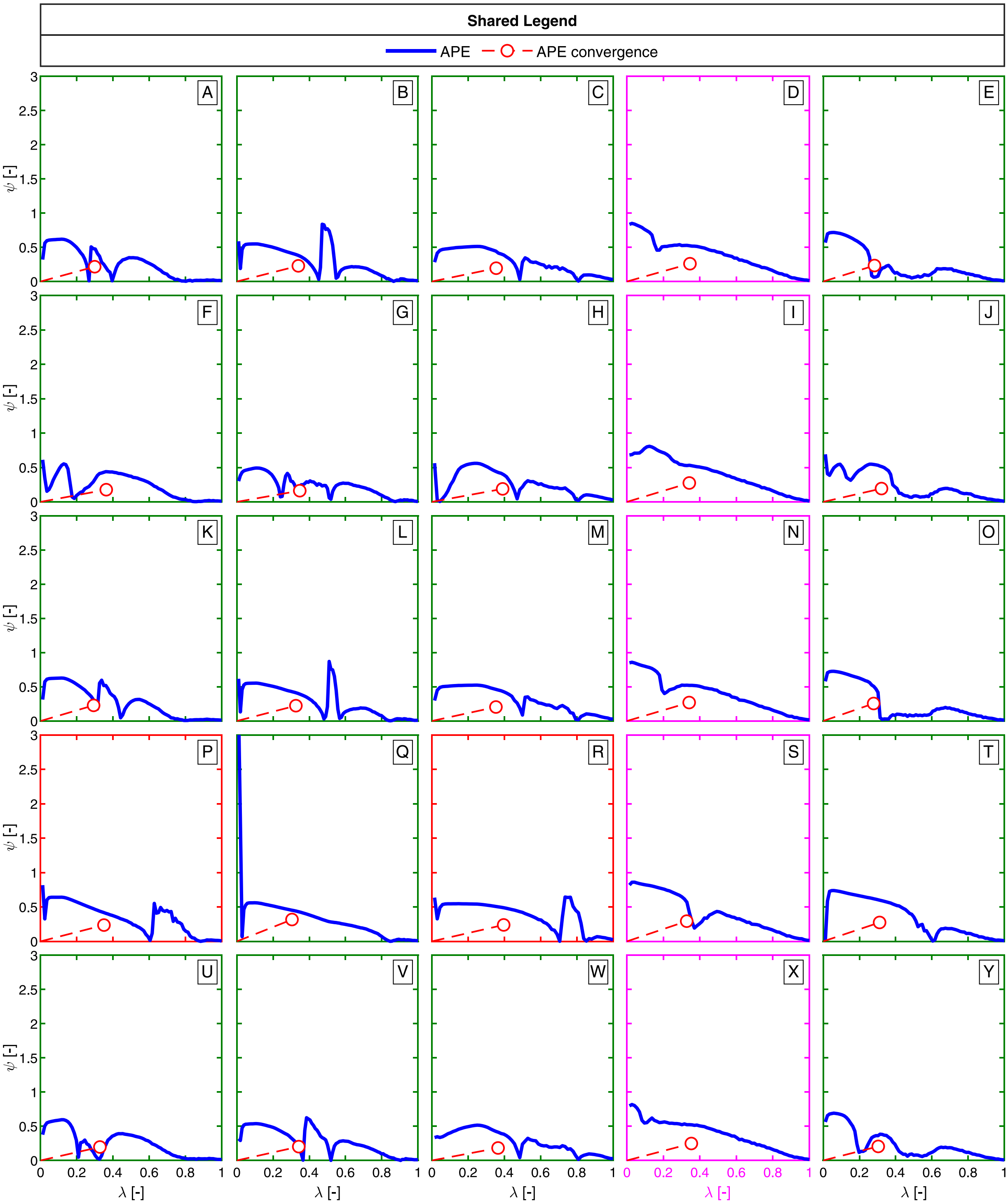


Fig. 10. Row index corresponds to the train RTF number, whereas column index to the validation RTF number. Graphs are identified by the letter in the box. Prognostic metrics (APE and C_{APE}) are represented for each train-validation combination. Axes are coloured in green for cases with $PH \geq 20\%$, in magenta for cases with $PH \geq 20\%$ but critical for a practical implementation, in red for cases with $PH < 20\%$. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

almost flat and underestimate the true *RUL* for a wide range of tool life. This behaviour is due to the fact that the algorithm tends to forget the initialisation and adapts to the faster degradation visible in the first region of the flank wear evolution. Even if the *PH*s are above 20% for these validation RTFs, these result are critical from a practical perspective. In fact, the algorithm predicts a null RUL for the major part of the tool life. This means that the algorithm considers the tool as failed earlier even with respect to the lower acceptability bound of the true *RUL*. Thus, the PF+P3 would suggested the operator to change the tool too early. Indeed these cases are considered as failed, despite the *PH* metric suggests the contrary. As a consequence, the authors believe that the field of prognosis metrics should be considered an area that has not been fully explored, yet. Therefore, researchers should put efforts to further develop off-line and on-line metrics for prognosis. Another consideration to be drawn to the reader attention regards the fact that in general, the shape described for cases A-G-M and Y is recurrent in most of the grid cases. Cases P and R are the only grid combinations that produce a *PH* below 20%. These cases exhibit similar characteristics associated with training and validation *VB* curves: the training *VB* curves share a similar initial region with the validation curves; in the final stage of the tool life, the validation *VB* curves show a rapid increase in slope. At the beginning, the algorithm starts to forget the initialisation and adapts to the available experimental data. This leads to an underestimation of the real RUL at the starting instants. Then, the algorithm tries to adapt the polynomials to the slower degradation in the region between 60% and 90% of tool life. Nevertheless, at this point of the RTF, the algorithm has not enough time to reach convergence with a *PH* beyond 20%. From *APE* plots (Fig. 10), this behaviour is highlighted by a longer region (between $\lambda = 0$ and about $\lambda = 0.6$) featuring higher errors with respect to the other cases. These cases exhibit *PH* values of 14% and 16% for cases P and R, and convergence values slightly higher with respect to the mean convergence on the grid, similarly to column 4 cases (Table 5).

When looking at one column per time in Fig. 9, it is worth nothing how the hybrid PF+P3 algorithm behaves in a similar way, independently from the training test. In fact, each column features comparable prediction shapes. Test Q is the only one showing a smoother adaptability rate, almost constant over time with respect to the other elements of the column. Despite the convergence metrics takes on the value of 0.44 (almost the maximum value among the other grid elements), *PH* is quite high and equal to 30%. Despite cases P and R do not meet the 20% threshold on *PH*, they seem to be isolated cases, and we consider critical, from the application point of view, the prediction over column 4 for the algorithm. Thus, the slowest combination of cutting parameters, i.e. low cutting speed and low feed rate, can be considered as the most critical.

As for the MLP case, the last analysis to be performed is the selection of the best cutting parameters for the initial training of the algorithm, which in turn means to look at the result grids row-wise. With this point of view, the hybrid PF+P3 scheme performs well in general. As shown in the previous paragraph each column of Fig. 9 features a similar prediction shape. Consequently, the training condition has a low influence on the prediction results. This is clear even from Fig. 10, *PH*s from Table 5 and convergences in Table 5. The only row resulting in poorer prediction outcomes is the fourth one, where two *PH*s fall below 20% of the tool life. Thus, the training condition to be avoided is surely the low cutting speed and feed rate combination.

3.2.3. *Comparison and discussion*

In this subsection, the results of the proposed hybrid schemes are compared with respect to standard applications of the PF and MLP. Furthermore, a comparison between the two hybrid solutions is performed, highlighting advantages and disadvantages of each solution. With regards to PF, two standard solutions were tested: a first one with an implementation that extended the Kalman filter (referred as *linearPF*) in [53]; a second one, which extended the linearPF solution with a cubic process equation (referred as *cubicPF*). The performances of all the algorithms are here compared on the basis of *PH*s. Table 6 summarises the *PH*s of the full set of algorithms in order to highlight differences between them.

First of all, the optimised linearPF and cubicPF solutions, although promising, were not as good as the ones for the conceived hybrid approach. Since in the cases of linearPF and cubicPF there was no initialisation, results were shown for all the validation tests only once. The adaptability of PF was kept and, for linearPF, *RUL* predictions reached the acceptability region near the end of the RTFs. Nevertheless, *RUL* confidence intervals were really large, leading to null probabilistic *PH*s for all the validation scenarios (Table 6). On the contrary, the optimal *cubicPF* results faced low adaptability to the true *RUL* of RTFs, with narrow confidence bounds. *cubicPF* exhibited two *PH* values equal to 21% out of 5. Nevertheless, the *RUL* behaviour for these conditions was similar to column 4 of PF+P3, resulting in critical *RUL* predictions, from the application point of view. On the other hand, the MLP becomes a static, non adaptive solution that predicts only the training *VB* curve. As a consequence, *APE*s are constant and have no convergence at all. Furthermore, no statistical description of *RUL* is provided. Indeed, in the case of MLP, only deterministic *PH*s [50] could be estimated and were reported in Table 6. Validation scenarios on the main diagonal of the table (same training and validation condition) provided exceptional results, with *PH*s equal to 100%. These results were expected since the MLP prediction curve coincides with the training curve. Nevertheless, considering the stochastic nature of tool wear, this is never the case, even for tests replicated in the same cutting conditions. An example can be found in the RTFs of the Prognostic Data Challenge 2010 dataset reported in [21], where different RTFs where performed at constant process parameters and the MLP would have failed to predict some cases, showing a 0% *PH*. More interesting is the behaviour on off-diagonal terms, where the MLP could predict correctly only pairs (train 1 — validation 3) and (train 3 — validation 1), which were featured by similar *VB* degradation curves. Therefore, the MLP could predict in advance 5 out of 5 main diagonal terms (100%, with a 100% *PH*), 2 out of 20 off-diagonal cases (10%, with a 100% *PH*), all accounting for 28% of the validation cases. In contrast, the two hybrid architectures could predict in advance the *RUL* of the validation tools in much more cases: the PF+MLP solution correctly predicted 5 out of 5 diagonal validations (100%, with a *PH* greater than 20%), 15 out of 20 off-diagonal cases (75%, with a *PH* greater than 20%), all accounting for 80% of the cases; the PF+P3 solution correctly predicts 4 out of 5 diagonal validations (100%, with a *PH* greater than 20%, but one critical for the application point of view), 14 out of 20 off-diagonal cases (90%, with a *PH* greater than 20%, but 4 critical for the application

Table 6
Prognostic horizons comparison. In green the PF+MLP results over 20%, in orange the PF+P3 results over 20% and in blue the MLP results over 20%. Critical results, from the application point of view, were not highlighted, regardless of the assumed *PH* value for the RTF.

PH [%]		Val. 1	Val. 2	Val. 3	Val. 4	Val. 5
Train 1	PF + MLP	**99%**	**52%**	**41%**	**32%**	**34%**
	PF + P3	**34%**	**27%**	**21%**	22%	**29%**
	MLP	**100%**	0%	**100%**	0%	0%
Train 2	PF + MLP	**66%**	**99%**	**53%**	0%	**42%**
	PF + P3	**33%**	**27%**	**21%**	23%	**28%**
	MLP	0%	**100%**	0%	0%	0%
Train 3	PF + MLP	**99%**	**55%**	**37%**	0%	**24%**
	PF + P3	**34%**	**28%**	**21%**	23%	**28%**
	MLP	**100%**	0%	**100%**	0%	0%
Train 4	PF + MLP	**30%**	**56%**	4%	**37%**	0%
	PF + P3	14%	**30%**	16%	22%	**29%**
	MLP	0%	0%	0%	**100%**	0%
Train 5	PF + MLP	**45%**	**39%**	10%	**22%**	**23%**
	PF + P3	**33%**	**27%**	**21%**	23%	**28%**
	MLP	0%	0%	0%	0%	**100%**
linear PF		0%	0%	0%	0%	0%
cubic PF		21%	0%	21%	0%	0%

Table 7
Qualitative comparison of algorithm characteristics. Initialisation: whether the algorithm is trained on a RTF or not. Adaptability: whether the algorithm can make use of online measurements or not. Speed: computational speed. *RUL* PDF: whether the algorithm can estimate the *RUL* PDF or not. Complexity: fitting model complexity. Reliability: capability of converging when varying RTF cutting parameters. Convergence speed: the higher the convergence speed, the higher the prognostics horizons.

Purely data-driven							
	Initialisation	Adaptability	Speed	*RUL* PDF	Complexity	Reliability	Conv. Speed
MLP	+	–	+ + +	–	+ +	–	+ +
Purely statistical							
	Initialisation	Adaptability	Speed	*RUL* PDF	Complexity	Reliability	Conv. Speed
linearPF	–	+	+ +	+	–	–	–
cubicPF	–	+	+	+	+	–	–
Hybrid framework							
	Initialisation	Adaptability	Speed	*RUL* PDF	Complexity	Reliability	Conv. Speed
PF+MLP	+	+ +	- -	+	+ +	+	+ +
PF+P3	+	+ +	–	+	+	+	+

point of view), all accounting for 72% of the cases. Indeed, it is evident that the adaptability feature of the hybrid framework allows to predict in advance more validation cases, at the expenses of the *PH* of each prediction. Moreover, it is necessary to compare the mean average prediction errors (when a tool reaches its end of life) for the three algorithms: PF+MLP scored on average 185 s on all the tests, PF+P3 16 s, while the MLP 815 s; when averaging only on unacceptable predictions (*PH* below 20%), PF+MLP scored 805 s, PF+P3 12 s, MLP 1129 s. These outcomes highlight the better performances of the proposed hybrid schemes.

When comparing the two proposed hybrid schemes, the PF+P3 algorithm correctly predicted more cases than the PF+MLP (from the *PH* perspective): 92% against 80%, respectively. Moreover, the average error on prediction is less for PF+P3 than for PF+MLP. This is mainly due to cases I and N, where the PF+MLP was not able to follow the new degradation rate. Nevertheless, on average the *PH*s of PF+MLP are significantly higher than those of the PF+P3 scheme (40% against 26%). Furthermore, 5 tests of the PF+P3 were considered as critical from the application point of view. Indeed, despite the lower number of cases with $PH < 20\%$, though still acceptable, the PF+MLP scheme provides greater flexibility in tool replacement, with a significant time lead ratio of 1.5. This also increases how much the operator can trust the predictions over the tool lifespan. At last, with a correct choice of the training RTF cutting parameters, i.e., high cutting speed, low feed rate, the PF+MLP was capable of correctly predicting all the cases (with a mean *PH* of 51% and a minimum *PH* of 32%). To summarise all the obtained results, a qualitative table (Table 7) comparing the algorithms qualities was shown.

4. Conclusions

The conceived approach allowed developing a Prognostics and Health Management framework for tool wear monitoring and prognosis. A hybrid solution was conceived as a combination of the statistical world and the data-driven world. From this perspective, the resulting methodology provided advantages with respect to the original separated components. Particle filter led to two benefits with respect to a standard data-driven implementation: the capability to predict the remaining useful life in a statistical fashion, providing its probability density function estimation; the adaptability of the data-driven model to the ongoing flank wear degradation, in opposition to its typical static implementation [4]. On the other hand, the data-driven models increased the flexibility of particle filter for remaining useful life prediction. In fact, in order to implement particle filter for online prognosis, a dynamical degradation model must be known for the particular application. A combined implementation of particle filter and a data-driven model does not require any dynamical formulation of the degradation process.

The main results obtained through the implementation of the proposed approach regard:

- a reduced experimental effort for the training dataset. The conceived approach was trained on a single run-to-failure experiment in order to test its adaptability. The algorithm was capable to follow online new degradation rates of cutting tools. In an industrial application scenario, the performances of the algorithm can be improved by adding new initialisations to the initial particles vectors, as long as cutting tools are worn out.
- generalisation with respect to different cutting conditions. The adaptability feature of the conceived approach allowed performing prognosis with fairly well performances on all the run-to-failures carried out with cutting parameter combinations not previously tested, limiting the needed training set dimension.
- estimation of the remaining useful life probability density function: specification required by international standards [6], but rarely satisfied in practical applications. This allowed computing more robust prognosis metrics (prognostics horizon, [50]), as well as to support maintenance decision making, since remaining useful life prediction bounds were available, too [4].
- good adaptability to other cutting parameters. One combination of train-validation set led the hybrid particle filter and multi-layer perceptron to excellent performances on cases not previously tested (100% predictions with PH above 20% of the tool life). The best scenario, from particle filter and multi-layer perceptron hybridisation, allowed reaching a minimum prognostics horizon of 32% (on average 51% and a maximum value of 99%). Three combinations led to optimal results (80% of cases with a prognostic horizon over 20%). It is reasonable to expect the algorithm to perform well within the cutting parameters suggested by the tool manufacturer. Thus, the algorithm is expected to work properly when no abrupt wear phenomena occur (i.e., when the tool is used within the suggested technological region of applicability). In this case, particular attention should be given in the choice of the training run-to-failure process parameters. High cutting speed and low feed rates should be chosen.
- particle filter with cubic polynomials provided the most reliable results from a metrics perspective, regardless of the training conditions. 92% of the cases led to a prognostic horizon over 20%. Nevertheless, 5 of these cases resulted in null remaining useful life predictions during a wide region of tool life, resulting critical from a practical perspective. Thus, only 72% of the cases were correctly predicted. Furthermore, the hybrid solution with particle filter and multi-layer perceptron was instead capable of making correct predictions with greater advance (mean prognostics horizon of 40% of tool life) with respect to the polynomial counterpart (mean prognostics horizon of 26% of tool life). Thus, the hybrid solution with particle filter and multi-layer perceptron offered a significant practical advantage.

Regarding future works, an interesting challenge would involve the integration of an adaptive real-time in-process indirect estimation algorithm with a pit-stop direct measurement technique such that Prognostics and Health Management could be responsive and more robust. The inclusion of process parameter data in the multi-layer perceptron architecture could increase the reliability of the predictions, as well.

Furthermore, the authors believe that the field of prognosis metrics should be considered an area of research not fully explored, yet. Therefore, researchers should put efforts to develop new off-line and on-line metrics for prognostics.

CRediT authorship contribution statement

Luca Bernini: Conceptualization, Formal analysis, Investigation, Methodology, Writing – original draft, Writing – review & editing. **Ugo Malguzzi:** Conceptualization, Data curation, Methodology, Software, Visualization. **Paolo Albertelli:** Conceptualization, Methodology, Supervision, Validation. **Michele Monno:** Supervision, Validation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

Acknowledgements

This work was supported by the Humans Hub project, Italy, funded by "Asse Prioritario I – Azione I.1.b.1.3 POR-FESR 2014–2020 Regione Lombardia" and a PhD grant (ID27 - "Prognostics and Health Management in Machine Tool and Manufacturing Industry") in DGR 769-2018 funded by Emilia-Romagna region, Italy.

References

[1] J. Wang, Y. Ma, L. Zhang, R.X. Gao, D. Wu, Deep learning for smart manufacturing: Methods and applications, J. Manuf. Syst. 48 (2018) 144–156, http://dx.doi.org/10.1016/j.jmsy.2018.01.003, URL: https://www.sciencedirect.com/science/article/pii/S0278612518300037, Publisher: Elsevier B.V..

[2] P. Mehta, P. Rao, Z. Wu, V. Jovanović, O. Wodo, M. Kuttolamadom, Smart manufacturing: state-of-the-art review in context of conventional and modern manufacturing process modeling, monitoring and control, in: International Manufacturing Science and Engineering Conference, 2018, http://dx.doi.org/10.1115/MSEC2018-6658.

[3] Y. Peng, M. Dong, M.J. Zuo, Current status of machine prognostics in condition-based maintenance: A review, Int. J. Adv. Manuf. Technol. 50 (1–4) (2010) 297–313, http://dx.doi.org/10.1007/s00170-009-2482-0.

[4] M. Baur, P. Albertelli, M. Monno, A review of prognostics and health management of machine tools, Int. J. Adv. Manuf. Technol. 107 (5) (2020) 2843–2863, http://dx.doi.org/10.1007/s00170-020-05202-3, Publisher: The International Journal of Advanced Manufacturing Technology.

[5] L. Bernini, D. Waltz, P. Albertelli, M. Monno, A novel prognostics solution for machine tool sub-units: The hydraulic case:, Proc. Inst. Mech. Eng. B (2021) http://dx.doi.org/10.1177/09544054211064682, Publisher: SAGE PublicationsSage UK: London, England.

[6] I.S.O. Central Secretary, Condition monitoring and diagnostics of machines - prognostics - part 1: general guidelines. Standard ISO 13381-1:2015. international organization for standardization, Geneva, CH, 2015.

[7] P. Stavropoulos, A. Papacharalampopoulos, E. Vasiliadis, G. Chryssolouris, Tool wear predictability estimation in milling based on multi-sensorial data, Int. J. Adv. Manuf. Technol. 82 (1–4) (2016) 509–521, http://dx.doi.org/10.1007/s00170-015-7317-6.

[8] S. Kurada, C. Bradley, A review of machine vision sensors for tool condition monitoring, Comput. Ind. 34 (1) (1997) 55–72, http://dx.doi.org/10.1016/s0166-3615(96)00075-9.

[9] X. Zhang, S. Wang, W. Li, X. Lu, Heterogeneous sensors-based feature optimisation and deep learning for tool wear prediction, Int. J. Adv. Manuf. Technol. 114 (9–10) (2021) 2651–2675, http://dx.doi.org/10.1007/S00170-021-07021-6/FIGURES/22, URL: https://link.springer.com/article/10.1007/s00170-021-07021-6, Publisher: Springer Science and Business Media Deutschland GmbH.

[10] X. Zhang, T. Yu, P. Xu, J. Zhao, In-process stochastic tool wear identification and its application to the improved cutting force modeling of micro milling, Mech. Syst. Signal Process. 164 (2022) 108233, http://dx.doi.org/10.1016/j.ymssp.2021.108233, URL: https://linkinghub.elsevier.com/retrieve/pii/S0888327021006051.

[11] M. Nouri, B.K. Fussell, B.L. Ziniti, E. Linder, Real-time tool wear monitoring in milling using a cutting condition independent method, Int. J. Mach. Tools Manuf. 89 (2015) 1–13, http://dx.doi.org/10.1016/j.ijmachtools.2014.10.011, Publisher: Elsevier.

[12] M. Kuntoglu, A. Aslan, D.Y. Pimenov, U.A. Usca, E. Salur, M.K. Gupta, T. Mikolajczyk, K. Giasin, W. Kaplonek, S. Sharma, A review of indirect tool condition monitoring systems and decision-making methods in turning: Critical analysis and trends, Sensors 21 (1) (2020) 108, http://dx.doi.org/10.3390/s21010108, URL: https://www.mdpi.com/1424-8220/21/1/108.

[13] International Standards, ISO 3685: Tool-Life Testing with Single-Point Turning Tools, ISO, 1993.

[14] M.T. García-Ordás, E. Alegre-Gutiérrez, R. Alaiz-Rodríguez, V. González-Castro, Tool wear monitoring using an online, automatic and low cost system based on local texture, Mech. Syst. Signal Process. 112 (2018) 98–112, http://dx.doi.org/10.1016/j.ymssp.2018.04.035, URL: https://www.sciencedirect.com/science/article/pii/S088832701830236X, Publisher: Academic Press.

[15] Y. Ao, G. Qiao, Prognostics for drilling process with wavelet packet decomposition, Int. J. Adv. Manuf. Technol. 50 (1–4) (2010) 47–52, http://dx.doi.org/10.1007/s00170-009-2509-6, URL: http://link.springer.com/10.1007/s00170-009-2509-6.

[16] P. Jia, Y. Rong, Y. Huang, Condition monitoring of the feed drive system of a machine tool based on long-term operational modal analysis, Int. J. Mach. Tools Manuf. 146 (March) (2019) 103454, http://dx.doi.org/10.1016/j.ijmachtools.2019.103454, Publisher: Elsevier Ltd.

[17] D.A. Tobon-Mejia, K. Medjaher, N. Zerhouni, CNC machine tool's wear diagnostic and prognostic by using dynamic Bayesian networks, Mech. Syst. Signal Process. 28 (2012) 167–182, http://dx.doi.org/10.1016/j.ymssp.2011.10.018.

[18] C. Letot, R. Serra, M. Dossevi, P. Dehombreux, Cutting tools reliability and residual life prediction from degradation indicators in turning process, Int. J. Adv. Manuf. Technol. 86 (1) (2015) 495–506, http://dx.doi.org/10.1007/S00170-015-8158-Z, URL: https://link.springer.com/article/10.1007/s00170-015-8158-z, Publisher: Springer.

[19] T. Benkedjouh, K. Medjaher, N. Zerhouni, S. Rechak, Health assessment and life prediction of cutting tools based on support vector regression, J. Intell. Manuf. 26 (2) (2015) 213–223, http://dx.doi.org/10.1007/s10845-013-0774-6, URL: http://link.springer.com/10.1007/s10845-013-0774-6.

[20] C. Zhang, X. Yao, J. Zhang, H. Jin, Tool condition monitoring and remaining useful life prognostic based on a wireless sensor in dry milling operations, Sensors 16 (6) (2016) 795, http://dx.doi.org/10.3390/s16060795, URL: http://www.mdpi.com/1424-8220/16/6/795.

[21] J. Yu, S. Liang, D. Tang, H. Liu, A weighted hidden Markov model approach for continuous-state tool wear monitoring and tool life prediction, Int. J. Adv. Manuf. Technol. 91 (1) (2017) 201–211, http://dx.doi.org/10.1007/s00170-016-9711-0.

[22] K. Javed, R. Gouriveau, X. Li, N. Zerhouni, Tool wear monitoring and prognostics challenges: a comparison of connectionist methods toward an adaptive ensemble model, J. Intell. Manuf. 29 (8) (2018) 1873–1890, http://dx.doi.org/10.1007/s10845-016-1221-2, URL: http://link.springer.com/10.1007/s10845-016-1221-2.

[23] B. Rabah, T. Benkedjouh, R. Said, Tool wear condition monitoring based on blind source separation and wavelet transform, in: M. Chadli, S. Bououden, S. Ziani, I. Zelinka (Eds.), Advanced Control Engineering Methods in Electrical Engineering Systems, Vol. 522, Springer International Publishing, Cham, 2019, pp. 377–389, http://dx.doi.org/10.1007/978-3-319-97816-1_29, URL: http://link.springer.com/10.1007/978-3-319-97816-1_29, Series Title, Lecture Notes in Electrical Engineering.

[24] B. Yan, L. Zhu, Y. Dun, Tool wear monitoring of TC4 titanium alloy milling process based on multi-channel signal and time-dependent properties by using deep learning, J. Manuf. Syst. 61 (2021) 495–508, http://dx.doi.org/10.1016/j.jmsy.2021.09.017, URL: https://linkinghub.elsevier.com/retrieve/pii/S027861252100203X.

[25] B. Guo, Q. Zhang, Q. Peng, J. Zhuang, F. Wu, Q. Zhang, Tool health monitoring and prediction via attention-based encoder-decoder with a multi-step mechanism, Int. J. Adv. Manuf. Technol. (2022) http://dx.doi.org/10.1007/s00170-022-09894-7, URL: https://link.springer.com/10.1007/s00170-022-09894-7.

[26] C.T. Wickramarachchi, T.J. Rogers, T.E. McLeay, W. Leahy, E.J. Cross, Online damage detection of cutting tools using Dirichlet process mixture models, Mech. Syst. Signal Process. 180 (2022) 109434, http://dx.doi.org/10.1016/j.ymssp.2022.109434, URL: https://linkinghub.elsevier.com/retrieve/pii/S0888327022005520.

[27] C. Drouillet, J. Karandikar, C. Nath, A.-C. Journeaux, M. El Mansori, T. Kurfess, Tool life predictions in milling using spindle power with the neural network technique, J. Manuf. Process. 22 (2016) 161–168, http://dx.doi.org/10.1016/j.jmapro.2016.03.010, URL: https://linkinghub.elsevier.com/retrieve/pii/S1526612516300123.

[28] P. Wang, Z. Liu, R.X. Gao, Y. Guo, Heterogeneous data-driven hybrid machine learning for tool condition prognosis, CIRP Ann. 68 (1) (2019) 455–458, http://dx.doi.org/10.1016/j.cirp.2019.03.007, URL: https://linkinghub.elsevier.com/retrieve/pii/S0007850619300083.

[29] R. Moliner-Heredia, I. Peñarrocha-Alós, J.V. Abellán-Nebot, Model-based tool condition prognosis using power consumption and scarce surface roughness measurements, J. Manuf. Syst. 61 (2021) 311–325, http://dx.doi.org/10.1016/j.jmsy.2021.09.001, URL: https://linkinghub.elsevier.com/retrieve/pii/S0278612521001874.

[30] X. Li, J. Zhou, H. Zeng, Y. Wong, G. Hong, An intelligent predictive engine for milling machine prognostic monitoring, in: 2006 IEEE International Conference on Industrial Informatics, IEEE, Singapore, 2006, pp. 1075–1080, http://dx.doi.org/10.1109/INDIN.2006.275766, URL: http://ieeexplore.ieee.org/document/4053539/.

[31] D. Shi, N.N. Gindy, Tool wear predictive model based on least squares support vector machines, Mech. Syst. Signal Process. 21 (4) (2007) 1799–1814, http://dx.doi.org/10.1016/j.ymssp.2006.07.016, URL: https://linkinghub.elsevier.com/retrieve/pii/S0888327006001725.

[32] X. Li, M.J. Er, B.S. Lim, J.H. Zhou, O.P. Gan, L. Rutkowski, Fuzzy regression modeling for tool performance prediction and degradation detection, Int. J. Neural Syst. 20 (05) (2010) 405–419, http://dx.doi.org/10.1142/S0129065710002498, URL: https://www.worldscientific.com/doi/abs/10.1142/S0129065710002498.

[33] K. Javed, R. Gouriveau, N. Zerhouni, R. Zemouri, X. Li, Robust, reliable and applicable tool wear monitoring and prognostic: Approach based on an improved-extreme learning machine, in: 2012 IEEE Conference on Prognostics and Health Management, IEEE, Denver, CO, USA, 2012, pp. 1–9, http://dx.doi.org/10.1109/ICPHM.2012.6299516, URL: http://ieeexplore.ieee.org/document/6299516/.

[34] M. Wang, J. Wang, CHMM for tool condition monitoring and remaining useful life prediction, Int. J. Adv. Manuf. Technol. 59 (5–8) (2012) 463–471, http://dx.doi.org/10.1007/s00170-011-3536-7, URL: http://link.springer.com/10.1007/s00170-011-3536-7.

[35] Y. Wu, G.S. Hong, W.S. Wong, Prognosis of the probability of failure in tool condition monitoring application-a time series based approach, Int. J. Adv. Manuf. Technol. 76 (1–4) (2015) 513–521, http://dx.doi.org/10.1007/s00170-014-6299-0, URL: http://link.springer.com/10.1007/s00170-014-6299-0.

[36] D. Liu, Z. Liu, J. Zhao, Q. Song, X. Ren, H. Ma, Tool wear monitoring through online measured cutting force and cutting temperature during face milling Inconel 718, Int. J. Adv. Manuf. Technol. (2022) http://dx.doi.org/10.1007/s00170-022-09950-2, URL: https://link.springer.com/10.1007/s00170-022-09950-2.

[37] J. Dou, S. Jiao, C. Xu, F. Luo, L. Tang, X. Xu, Unsupervised online prediction of tool wear values using force model coefficients in milling, Int. J. Adv. Manuf. Technol. 109 (3–4) (2020) 1153–1166, http://dx.doi.org/10.1007/s00170-020-05684-1, URL: https://link.springer.com/10.1007/s00170-020-05684-1.

[38] Q. Yang, K.R. Pattipati, U. Awasthi, G.M. Bollas, Hybrid data-driven and model-informed online tool wear detection in milling machines, J. Manuf. Syst. 63 (2022) 329–343, http://dx.doi.org/10.1016/j.jmsy.2022.04.001, URL: https://linkinghub.elsevier.com/retrieve/pii/S0278612522000528.

[39] O. Moldovan, S. Dzitac, I. Moga, T. Vesselenyi, I. Dzitac, Tool-wear analysis using image processing of the tool flank, Symmetry 9 (12) (2017) 296, http://dx.doi.org/10.3390/sym9120296, URL: http://www.mdpi.com/2073-8994/9/12/296.

[40] L. Fernández-Robles, L. Sánchez-González, J. Dí ez González, M. Castejón-Limas, H. Pérez, Use of image processing to monitor tool wear in micro milling, Neurocomputing 452 (2021) 333–340, http://dx.doi.org/10.1016/j.neucom.2019.12.146, URL: https://linkinghub.elsevier.com/retrieve/pii/S0925231220317501.

[41] W.-J. Lin, J.-W. Chen, J.-P. Jhuang, M.-S. Tsai, C.-L. Hung, K.-M. Li, H.-T. Young, Integrating object detection and image segmentation for detecting the tool wear area on stitched image, Sci. Rep. 11 (1) (2021) 19938, http://dx.doi.org/10.1038/s41598-021-97610-y, URL: https://www.nature.com/articles/s41598-021-97610-y.

[42] M. Cheng, L. Jiao, X. Shi, X. Wang, P. Yan, Y. Li, An intelligent prediction model of the tool wear based on machine learning in turning high strength steel, Proc. Inst. Mech. Eng. B 234 (13) (2020) 1580–1597, http://dx.doi.org/10.1177/0954405420935787.

[43] J. Zhang, B. Starly, Y. Cai, P.H. Cohen, Y.-S. Lee, Particle learning in online tool wear diagnosis and prognosis, J. Manuf. Process. 28 (2017) 457–463, http://dx.doi.org/10.1016/j.jmapro.2017.04.012, URL: https://linkinghub.elsevier.com/retrieve/pii/S1526612517300841.

[44] C. Sbarufatti, M. Corbetta, M. Giglio, F. Cadini, Adaptive prognosis of lithium-ion batteries based on the combination of particle filters and radial basis function neural networks, J. Power Sources 344 (2017) 128–140, http://dx.doi.org/10.1016/j.jpowsour.2017.01.105.

[45] F. Cadini, C. Sbarufatti, M. Corbetta, F. Cancelliere, M. Giglio, Particle filtering-based adaptive training of neural networks for real-time structural damage diagnosis and prognosis, Struct. Control Health Monit. 26 (12) (2019) 1–19, http://dx.doi.org/10.1002/stc.2451.

[46] D.L. Elliott, A Better Activation Function for Artificial Neural Networks, Technical Report, 1993.

[47] G. Cybenko, Approximation by superpositions of a sigmoidal function, Math. Control Signals Systems 2 (4) (1989) 303–314, http://dx.doi.org/10.1007/BF02551274, URL: http://link.springer.com/10.1007/BF02551274.

[48] D. Nguyen, B. Widrow, Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights, in: 1990 IJCNN International Joint Conference on Neural Networks, Vol.3, 1990, pp. 21–26, http://dx.doi.org/10.1109/IJCNN.1990.137819.

[49] D. Wang, P.W. Tse, Prognostics of slurry pumps based on a moving-average wear degradation index and a general sequential Monte Carlo method, Mech. Syst. Signal Process. 56 (2015) 213–229, http://dx.doi.org/10.1016/j.ymssp.2014.10.010, Publisher: Elsevier.

[50] A. Saxena, J. Celaya, B. Saha, S. Saha, K. Goebel, Metrics for offline evaluation of prognostic performance, Int. J. Prognost. Health Manag. 1 (1) (2010) 2153–2648, http://dx.doi.org/10.36001/ijphm.2010.v1i1.1336.

[51] D.C. Montgomery, Design and Analysis of Experiments, Vol. 2, John Wiley & Sons, Inc., 2012, http://dx.doi.org/10.1198/tech.2006.s372, URL: http://cataleg.uab.cat/record=b1764873~S1*cat, Publication Title, Design ISSN: 00401706.

[52] D. Johansson, S. Hägglund, V. Bushlya, J.-E. Ståhl, Assessment of commonly used tool life models in metal cutting, Procedia Manuf. 11 (2017) 602–609, http://dx.doi.org/10.1016/j.promfg.2017.07.154, URL: https://linkinghub.elsevier.com/retrieve/pii/S235197891730358X.

[53] J.P.P. Gomes, B.P. Leão, W.O. Vianna, R.K. Galvão, T. Yoneyama, Failure prognostics of a hydraulic pump using Kalman Filter, in: Proceedings of the Annual Conference of the Prognostics and Health Management Society 2012, PHM 2012, Vol. 4, no. 1, 2012, pp. 464–468, ISBN: 9781936263059.