Journal of Manufacturing Systems 59 (2021) 522–534

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

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

Physics-guided logistic classification for tool life modeling and process parameter optimization in machining[z.star;]

Jaydeep Karandikar [a,*], Tony Schmitz [a,b], Scott Smith [a]

[a] Manufacturing Science Division, Oak Ridge National Laboratory, Oak Ridge, TN, 37830, USA
[b] Department of Mechanical, Aerospace, and Biomedical Engineering, University of Tennessee, Knoxville, TN, 37996, USA

ARTICLE INFO

Keywords:
Machine learning
Classification
Machining
Tool life
Optimization
Uncertainty

ABSTRACT

This paper describes a physics-guided logistic classification method for tool life modeling and process parameter optimization in machining. Tool life is modeled using a classification method since the exact tool life cannot be measured in a typical production environment where tool wear can only be directly measured when the tool is replaced. In this study, laboratory tool wear experiments are used to simulate tool wear data normally collected during part production. Two states are defined: tool not worn (class 0) and tool worn (class 1). The non-linear reduction in tool life with cutting speed is modeled by applying a logarithmic transformation to the inputs for the logistic classification model. A method for interpretability of the logistic model coefficients is provided by comparison with the empirical Taylor tool life model. The method is validated using tool wear experiments for milling. Results show that the physics-guided logistic classification method can predict tool life using limited datasets. A method for pre-process optimization of machining parameters using a probabilistic machining cost model is presented. The proposed method offers a robust and practical approach to tool life modeling and process parameter optimization in a production environment.

1. Introduction

Tool life is one of the significant limitations to machining productivity. The wear mechanisms that lead to tool wear are abrasion, adhesion, diffusion, and attrition [1,2]. In a production environment, pre-process prediction of tool life as a function of process parameters for a given tool-material combination is important for process parameter optimization. This is because there exists a trade-off between machining time per part and tool life; this is due to a reduction in tool life with an increase in material removal rate. This is illustrated in Fig. 1 which shows the dependence of machining cost per part with cutting speed. At low cutting speeds, machining time dominates the total machining cost per part due to low material removal rates. An increase in cutting speed increases the material removal rate, which reduces the machining time. However, tool life reduces with cutting speed. Therefore, at high cutting speeds, tool life and tool change time dominate the machining cost per part due to low tool life requiring multiple tool changes. The optimal cutting speed balances the trade-off between machining time and tool life.

There have been three distinct approaches for modeling tool wear and tool life presented in the literature: empirical [3–5], physics-based [6–9], or stochastic methods using probabilistic and reliability analyses [10–15]. The three approaches need model coefficient calibration through tool wear experimentation. For example, the wear model proposed by Usui et al. [7,8], requires experimentation for calibration of the model coefficients A, and B. Response surface methods or empirical models need experiments to calibrate the model coefficients. Similarly, probabilistic methods such as reliability analysis [11,12], or Bayesian updating [13,11–15] need experimentation to calculate the probability distributions for tool life as a function of process parameters. This presents a significant limitation in implementing any approach for tool life modeling in a production environment. Although the predictive capabilities for these approaches are good after the model coefficients are calibrated, they cannot be generalized for different tool-material

* Corresponding author.
E-mail address: karandikarjm@ornl.gov (J. Karandikar).

https://doi.org/10.1016/j.jmsy.2021.03.025
Received 2 February 2021; Received in revised form 29 March 2021; Accepted 29 March 2021
Available online 10 April 2021

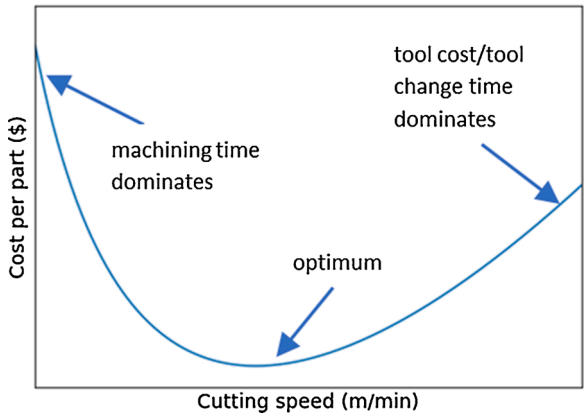

Fig. 1. Machining cost per part as a function of cutting speed.

combinations. The calibrated model coefficients and the associated tool life predictions are specific to the tool and material combination tested, so the calibration experiments must be repeated for all unique tool-material combinations. Furthermore, the model coefficients are only valid in the range of the test parameters. The extensive tool life experimentation needed to calibrate the tool life predictive models is expensive and time-consuming and, therefore, infeasible in a production environment for many tool-material combinations.

This paper presents the application of machine learning (ML) for tool life modeling using production parts tool wear data as a practical method for predicting tool life and process parameter optimization in a production environment. Machine learning methods enable the identification of patterns from underlying data [16,17]. The concept implemented here is to consider the production of real parts as tool wear experiments and use the collected tool wear data in a machine learning framework for predicting tool life. The advantage of the proposed approach is that separate tool wear experiments need not be performed to calibrate model coefficients. Instead, the tool wear information collected during part production can be used to model tool life. Furthermore, data collected during the production of actual parts is more representative of the tool performance as opposed to tool wear experiments performed in a controlled laboratory environment on a test workpiece. The proposed approach is motivated by two challenges. First, in a production environment, tool wear can typically be measured only at the time of tool change resulting in a single data point on the tool wear and machining time curve. The exact time for the end of tool life (when the tool wear reaches the predetermined wear limit) is not available. Therefore, a regression on tool life values at different process parameters cannot be performed. Second, process parameters in a production environment tend to be clustered at a few preferred parameters. These parameters are selected by the programmer based on experience, tool manufacturer recommendations, or other ad hoc resources. As a result, the machine learning classification model must be extrapolated outside the input data range to predict tool life and optimize process parameters. To address these challenges, this paper presents a physics-guided logistic machine learning classification method for modeling tool life through input variable transformation. Since the exact time for the end of tool life cannot be typically measured in a production environment, tool life is modeled with machine learning classification. Two classes are defined: tool not worn (class 0) and tool worn (class 1), based on the measured tool wear level. If the predefined wear limit is greater than the measured tool wear, the tool is not worn (class 0); otherwise, it is worn (class 1). For a given set of process parameters, the predicted tool life is the ML classification decision boundary in time that separates the two classes.

There have been many efforts in the literature on the application of data-driven and ML methods for tool life monitoring. Wang et al. used a Bayesian learning approach for an event-driven tool condition monitoring approach; the Bayesian method was used to predict the remaining useful life using monitored sensor data [18]. Liu et al. used spindletorque to monitor tool wear for repitive operations using a similarity metric [19]. Wang et al. developed a physics-guided neural network using force and vibration to predict in-process tool wear [20]. Corne et al. used spindle power in drilling with neural network to monitor tool wear [21]. Additional examples in the literature on the application of machine learning for tool wear monitoring are provided by references [18–24]. As seen in references [18–24], the objective of using ML for tool life monitoring is the real-time prediction of tool wear or remaining useful tool life using in-process sensor data such as cutting force or vibration. The application of ML for tool life monitoring does not address the pre-process selection of process parameters and the associated expected tool life. A pre-process modeling of tool life as a function of process parameters enables a pre-process optimization for the cycle time and tool life trade-off presented in Fig. 1. The application of ML classification for pre-process modeling of tool life using wear data has not been presented in the literature. The method proposed in this paper differs from the tool wear monitoring using ML studies presented in the literature as follows. First, the objective is a pre-process prediction of tool life as a function of process parameters. Second, the proposed approach uses direct tool wear measurements completed after the tool is replaced and does not need any additional sensors to monitor the in-process tool wear. As a result, the method can be used in conjunction with the real-time in-process tool wear monitoring methods presented in the literature. The method described in the paper can be used for a pre-process selection of optimal process parameters and the expected tool life. The methods in references [18–24] can be subsequently used for in-process tool wear monitoring to account for stochastic tool wear. This paper expands on the previous work by the authors [25] by validating the physics-guided logistic model results through experimental validation in milling and providing interpretability for the model results through a direct comparison with the well-known Taylor tool life model. Additionally, a method for optimizing the process parameters using the logistic model predictions and a probabilistic cost model is presented.

The proposed solution has three main advantages. First, modeling tool life using the shop floor tool wear data eliminates the need for expensive tool life experimentation. Second, using the physics-guided logistic model, the process parameter-dependent tool life model can be quickly identified using limited datasets collected from production parts for each tool-material combination. The model can be subsequently used to determine the optimal machining parameters to minimize machining cost per part. Third, as noted, the method does not need in-process sensors to monitor tool wear. The tool wear data can be collected by measuring tool wear at the time of tool replacement using a low-cost microscope. The remainder of the paper is organized as follows. Section 2 describes milling experimental tool life results at different cutting speeds and a method to simulate production tool wear data as a function of cutting speed. Section 3 shows tool life modeling results for the neural network classification method and presents the motivation for a physics-guided machine learning classification method. Section 4 describes the physics-guided logistic classification method using a logarithmic transformation of the input variables. A method for model interpretability is described and the model predictions are validated using experimental results. Section 5 describes a modified cost equation and the method to select optimal process parameters using the physics-guided logistic model tool life predictions. A discussion is provided in Section 6 which presents a method to augment imbalanced datasets, a hybrid logistic classification model that combines linear and logarithmic inputs, and a method to extend the model to multiple variables. Conclusion and future work are presented in Section 7.

2. Experimental results

This section describes the experimental results used to simulate tool wear data from a production environment. Tool wear tests were

completed on 1018 steel workpiece material in down-milling. The tool was a 19.05 mm diameter single-insert endmill (Kennametal KICR073SD30333C) with a square uncoated carbide insert (Kennametal 107,888,126 C9 JC). Three tool wear tests were performed at 149.6 m/min (2500 rpm) and 299.2 m/min (5000 rpm). The feed per tooth was 0.06 mm, the axial depth of cut was 3 mm, and the radial depth of cut was 4.7 mm (25 % radial immersion), respectively. The insert wear status was measured at regular intervals using a handheld microscope. Fig. 2 shows the progression of flank wear as a function of cut time for the first test at 149.6 m/min. No rake wear was observed. Fig. 3 shows the maximum flank wear width (*FWW*) as a function of cut time at 149.6 m/min and 299.2 m/min. The end of tool life was reached when the maximum *FWW* was 0.3 mm [26]. If the flank wear width exceeded 0.3 mm at the time of measurement, the tool life was calculated by linear interpolation between the two adjacent *FWW* measurements. The tool life results are listed in Table 1. As seen from Fig. 3, the tool life reduced with spindle speed. Tool life results listed in Table 1 show variation in tool life from repeat tests.

The experimental results listed in Table 1 were used to simulate representative tool wear data collected from production parts in a production environment. The tool life for both spindle speeds was modeled as a normal distribution. The mean tool life, denoted by t_m, and the standard deviation of tool life, denoted by t_s, was calculated from the three test results at each speed shown in Table 1. The production environment data was generated at each cutting speed as follows. First, a tool life value was sampled from the normal distribution given by the Table 1 test results at the selected cutting speed. Second, the tool wear measurement time was set by sampling from a uniform distribution in the interval $[0, t_m + 3 \times t_s]$. Third, the tool life sample was compared to the tool wear measurement time sample. If the tool life sample is greater than the tool wear measurement time, the tool was considered not worn (class 0). Otherwise, the tool was worn (class 1). Fig. 4 shows the simulated data points; the left panel displays 10 data points at 149.6 m/min and the right panel displays the data points at 299.2 m/min. In Fig. 4, blue denotes tool not worn (class 0) and red denotes tool worn (class 1). The data shown in Fig. 4 can be interpreted as data collected for a tool-material combination from multiple machines making different parts in a production environment. The two cutting speeds, 149.6 m/min (2500 rpm) and 299.2 m/min (5000 rpm), can be considered as preferred speeds for the tool-material combination. As noted, in a typical production environment, the cutting speed is selected by the programmers based on experience, tool supplier recommendations, or other resources. Since the tool life is not known, a tool is used for an interval informed by experience, part geometry, and/or volume removed for a given material removal rate. As seen in Fig. 4, the data has more points where the tool is not worn (16) than when the tool is worn (4). This is illustrative of the data collected from different production parts during operation in a production environment. The simulated data is not perfectly separable because the tool worn and the tool not worn data points overlap. This accounts for the tool life non-repeatability (and subsequent uncertainty) observed in Table 1. Note that the objective of the data simulation is to test the ML models on a dataset that mimics data collected from a typical production environment.

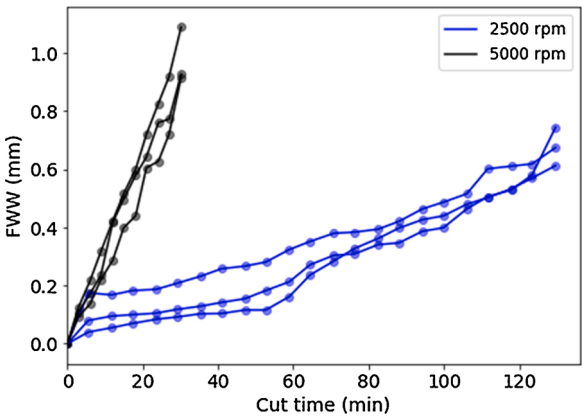

Fig. 3. Flank wear width as a function of cut time at 149.6 m/min (2500 rpm) and 299.2 m/min (5000 rpm).

Table 1
Experimental tool life results at 149.6 m/min and 299.2 m/min.

Experiment number	Cutting speed (m/min)	Spindle speed (rpm)	Tool life (min)
1	149.6	2500	50.1
2	149.6	2500	68.5
3	149.6	2500	72.0
4	299.2	5000	11.5
5	299.2	5000	8.5
6	299.2	5000	9.5

3. Neural network classification for tool life modeling

Section 2 described the experimental results and a method to simulate production environment data. In this section, the simulated data is used to train a neural network (NN) classifier. Recall that the tool life is given by the classifier decision boundary which separates the tool not worn (class 0) and tool worn (class 1) classes. The NN classifier consists of multiple layers; the first layer is an input layer, the final layer is an output layer, and they are separated by one or more hidden layers. Each layer has nodes (or neurons). There is an activation function for each node in the hidden layers and the output layer that transforms the sum of each output of the previous layer nodes multiplied by a weight and a bias term, b [27]. This is shown in Eq. 1, where a is the output of a node, g is the activation function, w is the weight, b is the bias, n is the number of

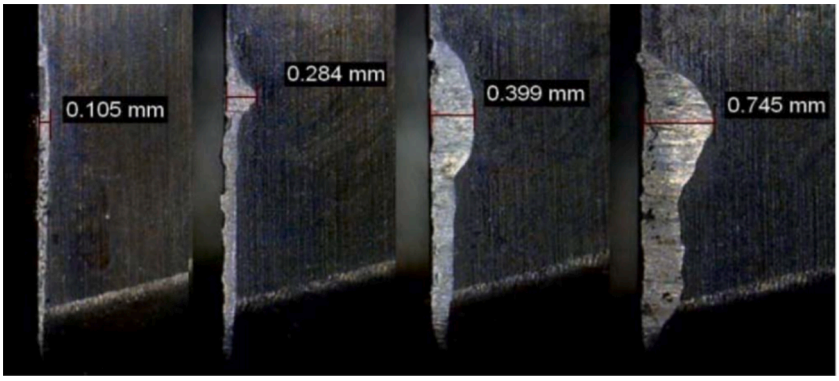

Fig. 2. Progression of flank wear as a function of cut time; from left to right, the cut time was {23.6, 64.5, 99.8, and 129.6} minutes.

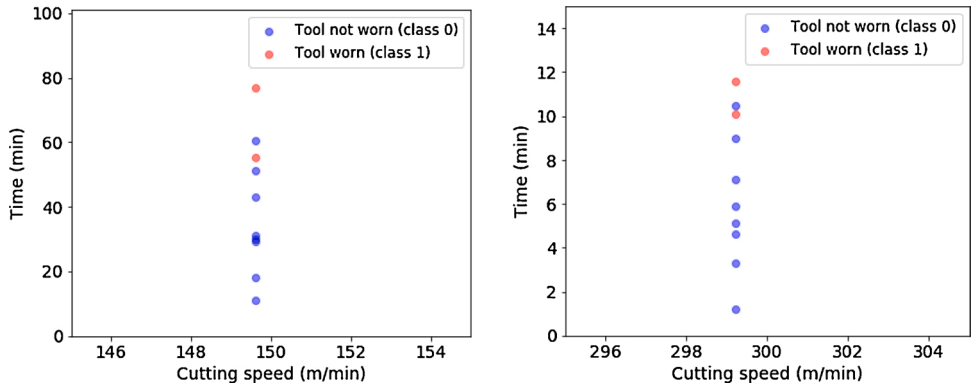

Fig. 4. Simulated production tool wear data from laboratory tests at 149.6 m/min (left panel) and 299.2 m/min (right panel). Note the difference in scales for the vertical axis.

nodes in layer l, and j and i correspond to individual nodes in layers l and l-1, respectively.

$$a_j^{[l]} = g^{[l]}\left(\sum_{i=1}^{n^{[l-1]}} w_{j,i}^{[l-1]} a_i^{[l-1]} + b_j^{[l]}\right) \tag{1}$$

The hyperbolic tangent activation function is one of the most common type of activation function. See Eq. 2, where z is the total output given by the bracket term in Eq. 1.

$$g(z) = \frac{\sinh z}{\cosh z} = \frac{e^z - e^{-z}}{e^z + e^{-z}} \tag{2}$$

The activation function of the output layer is the softmax (also called normalized exponential) function for binary classification problems. The softmax function normalizes the output of each node by the sum of the output for all nodes. The output of the softmax function can then be interpreted as the probability that the sample belongs to a particular class. Eq. 3 shows the softmax activation function.

$$g(z) = \frac{e^z}{\sum_{i=1}^{n} e^z} \tag{3}$$

In Eq. 3, i denotes the number of nodes in the output layer. The weights and bias values for each layer are learned by minimizing the binary cross-entropy loss function shown in Eq. 4, where L denotes the loss, m denotes the total number of data points used for training the network, y_k is the true class for sample k, and $\hat{y}_k$ is the predicted class for sample k.

$$L = -\frac{1}{m}\sum_{k=1}^{m}[y_k \cdot \log(\hat{y}_k) + (1 - y_k) \cdot \log(1 - \hat{y}_k)] \tag{4}$$

The loss function can be regularized by adding a term $\frac{\alpha}{2m}\left\|w^{[l]}\right\|^2$ where $\|\cdot\|^2$ is the L2 norm. The regularization parameter mitigates the overfitting of the weights by assigning a penalty to the loss function; this prevents the weights from taking large values [27]. The loss function is minimized through gradient descent-based methods [27]. The idea is to take a step in the direction where the local gradient in the loss function is the highest. This is calculated using a derivative of the loss function through backward propagation starting with the final layer. For brevity, the mathematical details of the optimization algorithms and backward propagation are not included in this paper. The reader is referred to references [27,28] for details on the training methods for the NN classifier. The NN was trained using the data shown in Fig. 4 with cutting speed and time as the inputs. The training was implemented with the scikit-learn library in the Python programming language [29,30]. The simulated data points were shuffled and separated into a training set (15 data points) and a test set (5 data points). The training set included three of the four tool worn (class 1) data points from Fig. 4 and the test set included the remaining tool worn (class 1) data point. The structure of the neural network was optimized by considering the number of hidden layers and the number of nodes in each layer as hyperparameters. The number of candidate hidden layers was taken as either 1, 2, or 3 with the number of nodes being either 20 or 50. In addition, a regularization parameter, α, was also considered as a hyperparameter. The hyperparameters were tuned using three-fold stratified cross-validation over the training set using the f1-score as a scoring metric. The Adam optimization algorithm was used for training the neural network [27]. Although a five-fold approach is typically recommended for cross-validation, three-fold was chosen in this case due to the small training dataset. The hyperparameter tuning was performed using an exhaustive grid search by the scikit-learn GridSearchCV function [30]. The f1-score was used as a metric instead of accuracy since the dataset is imbalanced; using accuracy as a metric can produce a less robust result. For example, a model that predicts tool not worn (class 1) for all test points would have an accuracy of 0.8. The optimal hyperparameters maximize the average f1-score from all three folds. The optimal NN structure was found to be three hidden layers with 20, 50, and 20 nodes, respectively. The regularization parameter was 0.0177. Fig. 5 shows the probability of tool worn (left panel) and the decision boundary (right panel) as a function of cutting speed using the trained NN, where the probability of tool worn (class 1) for the given {cutting speed, time} combination is identified by the gray scale. The decision boundary (Fig. 5 right panel) was based on a threshold probability for tool worn equal to 0.5. The probability of tool worn is determined from the NN output for each cutting speed and time combination discretized with an interval of 1 m/min in cutting speed and 0.1 min in time. The training data points are represented by the lighter color and the test data points by the darker colors. As seen from Fig. 5, the NN classifies the five test points accurately.

3.1. Motivation for a physics-guided classification method

As noted, in a production environment, the objective of a classifier is to predict tool life as a function of cutting speed, which can subsequently be used for pre-process optimization of machining parameters to minimize the machining cost per part. To validate the result from the trained NN network, an experiment was performed at an intermediate cutting speed of 224.4 m/min (3750 rpm); the experimental procedure is described in Section 2. A tool life of 35.5 min was observed based on the 0.3 mm maximum *FWW* criterion. Fig. 6 shows the probability of tool worn (class 1) for the trained NN at 224.4 m/min. From a tool life modeling standpoint, the probability of tool worn can also be interpreted as the probability that the tool life is less than or equal to the time value on the horizontal axis. The experimental result of 35.5 min is shown as a dashed vertical line. Based on the single experimental result of 35.5 min, the tool life is slightly underpredicted by the NN; as seen

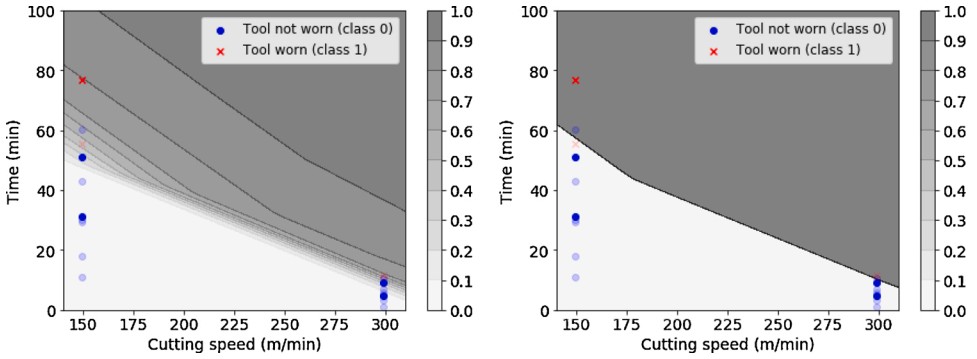

Fig. 5. Probability of tool worn (left panel) and decision boundary (right panel) as a function of cutting speed from the trained NN model; the train data points are shown in light and the test data points are shown in dark.

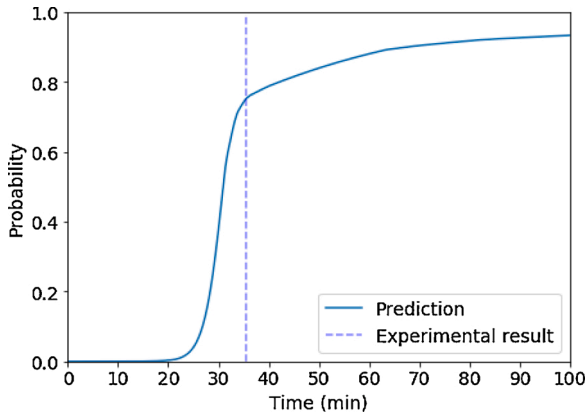

Fig. 6. Predicted probability of tool worn at 224.4 m/min from the trained NN model (solid line) and the experimental result (dashed line).

from Fig. 6, the predicted probability of tool worn at 35.5 min was 0.75. However, since tool life is non-repeatable, Fig. 6 shows that the trained NN can capture the experimental tool life in the predicted distribution. This is because, as seen From Fig. 6, the probability of tool life to be between 30 min and 40 min is 0.58.

To evaluate the performance of the NN model outside the training data range, a prediction for the probability of tool worn was made in the range from 89.7 m/min (1500 rpm) to 448.8 m/min (7500 rpm). Validation tool wear experiments were performed at 89.7 m/min and 448.8 m/min. Tool life values of 255.3 min and 3.3 min were observed at 89.7 m/min and 448.8 m/min, respectively. Fig. 7 shows the probability of tool worn from the trained NN with cutting speed in the range from 89.7 m/min to 448.8 m/min. As seen from Fig. 7, the NN performs poorly when extrapolating the results outside the input data range. The NN predicts a tool life smaller than 0 min at cutting speeds greater than 350 m/min. At cutting speeds less than 149.6 m/min, the NN cannot incorporate the non-linear increase in tool life, and therefore, under-predicts the tool life significantly at 89.7 m/min. This is seen in Fig. 7 where the NN predicts a tool life of 89.2 min for a 0.5 probability of tool worn (class 1) at 89.7 m/min. This is expected since the NN is a data-driven approach and cannot generalize at parameters beyond the training data set. This is especially true when training a NN with a small data set. Results shown in Fig. 7 demonstrate the need for a physics-guided ML classification method for tool life modeling. With a small dataset, traditional machine learning methods cannot generalize beyond the training data and cannot consider the underlying physics or constraints for a given process. Note that if the number of tool wear data points were large (>100) and uniformly distributed in the entire cutting speed range, NN (as well as other ML algorithms) will be able to model the underlying tool life accurately.

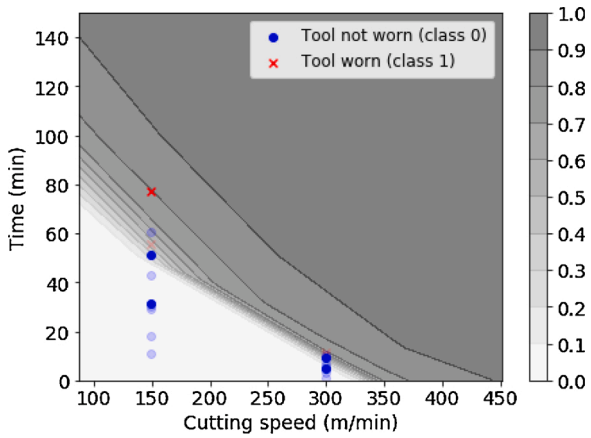

Fig. 7. Probability of tool worn from the trained NN model as a function of cutting speed; the cutting speed range is extended from 89.7 m/min (1500 rpm) to 448.8 m/min (7500 rpm).

4. Physics-guided logistic classification for tool life modeling in milling

This section describes a physics-guided logistic classification for tool life modeling. A logistic classifier calculates the probability of class membership given input data using the sigmoid function [28] shown in Eq. 5 and Eq. 6, where p is the probability, k is the number of input features x, y is the class membership, θ_n are the logistic model parameters, and $g(x)$ is a linear combination of the k input features.

$$p(y = 1x, \theta) = \frac{1}{1 + e^{-g(x)}} \tag{5}$$

$$g(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots \theta_n x_k \tag{6}$$

The sigmoid function converts the linear output to a value between zero and one. In training the logistic classifier, the parameter θ_n is calculated by minimizing the binary cross-entropy loss function shown in Eq. 4. The logistic classifier was trained using the simulated data shown in Fig. 4. As noted, the simulated data points were shuffled and separated into a training set (15 data points) and a test set (5 data points). As with the NN, the inputs to the logistic classifier are the cutting speed and time. The logistic regularization parameter, λ, was treated as a hyperparameter and tuned using three-fold cross-validation with the f1-score as a metric. This was done with the GridSearchCV function in scikit-learn with a range of λ values from 0 to 10^{10} [30]. Fig. 8 displays the probability of tool worn (class 1) as a function of cutting speed from the trained logistic classifier. The probability of tool worn shown in Fig. 8 is calculated from the trained logistic model for

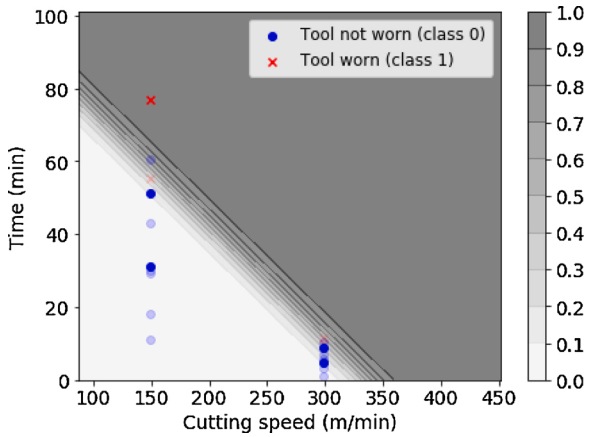

Fig. 8. Probability of tool worn as a function of cutting speed from the trained logistic model; the train data points are shown in light and the test data points are shown in dark.

each cutting speed and time combination discretized with an interval of 1 m/min in cutting speed and 0.1 min in time. In addition to the extrapolation challenges faced by the NN, the logistic classifier is also linear and cannot capture the non-linear relationship between tool life and cutting speed.

Even though the logistic classifier is linear, it has three characteristics that make it suitable for modeling tool life. First, as shown in Eqs. 5 and 6, the logistic classifier is easy to interpret, and implement. Second, the logistic classifier provides a probabilistic view of class predictions. This is important for tool life modeling due to the non-repeatable and stochastic behavior of tool life. Third, the logistic classifier can be used to model complex non-linear decision boundaries through a simple transformation of the input features. For tool life modeling, the input features can be modified based on the knowledge of tool life in machining. The following assessments can be made from machining process knowledge regarding the relationship between tool life and cutting speed in machining. First, it is known that tool life reduces non-linearly with cutting speed [1,2]. Tool wear is a temperature-driven process and the cutting temperature increases with cutting speed. It has been established that this relationship can be described by a power law [1,2,31–33]. The power law relationship between tool life and cutting speed is given by the Taylor tool life equation [31]:

$$VT^n = C \tag{7}$$

In Eq. 7, n and C are coefficients that depend on the tool-workpiece combination, T is the tool life, and V is the cutting speed. Taking the logarithm of the Eq. 7 Taylor tool life equation gives Eq. 8.

$$\log(V) + n\log(T) = \log(C) \tag{8}$$

$$\log(T) = -\frac{1}{n}\log(V) + \frac{\log(C)}{n} \tag{9}$$

By rearranging terms in Eq. 9, it is seen that the Taylor tool life equation is linear in the logarithmic space. This behavior can be incorporated in the linear logistic classifier by performing a logarithmic transformation of input variables: cutting speed and time. This helps mimic the behavior of tool life described by the Taylor tool life equation and incorporates process knowledge in the logistic model. With this transformation, the classifier's decision boundary is linear in the logarithmic domain and follows the non-linear power-law relationship in the non-transformed original domain. The logistic classification model was trained with the logarithmic values of cutting speed and time shown provided in Fig. 4. Fig. 9 shows the probability of tool worn (class 1) from the trained logistic model using a logarithmic transformation of the input variables; the left panel shows the probability on the logarithmic scale for cutting speed and time and the right panel shows the original values. As seen from Fig. 9, the logarithmic transformation of the input variables enables the logistic classifier to mimic the non-linear exponential decrease in tool life with cutting speed. As a result, the logistic classifier tool life prediction can be extrapolated outside the data space since the exponential decrease in tool life is explicitly modeled in the logistic classifier. The tool life prediction increases exponentially at cutting speeds smaller than 149.6 m/min and does not become zero at cutting speeds greater than 299.2 m/min. As noted, from a tool life modeling standpoint, the probability of tool worn (class 1) is also the probability that the tool life is less than or equal to the time value on the horizontal axis. Fig. 10 shows the probability of tool worn (class 1) as a function of time and the experimental result at 89.8 m/min (1500 rpm) (top left panel), 224.4 m/min (3750 rpm) (top right panel), and 448.8 m/min (7500 rpm) (bottom left panel). As seen from Fig. 10, the experimental data lies in the predicted distribution of worn tool class at all three cutting speeds. Note that the tool life is underpredicted at 224.4 m/min since the probability of tool worn at 35.5 min is 0.9. However, as noted, since tool life is non-repeatable, tool life is better characterized by the probabilistic output of the logistic classifier (as opposed to a deterministic value given by the decision boundary of the classifier).

Machine learning (ML) models are generally considered "black box" where the ML model provides the relationship, but its structure is not interpretable by humans. As noted, cutting speed and time inputs for the physics-guided logistic classifier were transformed into the logarithmic domain. For a logistic classifier, tool life is given by the decision boundary between the two classes which satisfies the equation $g(x) = 0$ as shown in Eq. 5 and Eq. 6. The logistic decision boundary for the log-transformed variables can be rearranged as shown in Eq. 10 and Eq. 11.

$$\theta_0 + \theta_1\log(V) + \theta_2\log(T) = 0 \tag{10}$$

$$\log(T) = -\frac{\theta_1}{\theta_2}\log(V) - \frac{\theta_0}{\theta_2} \tag{11}$$

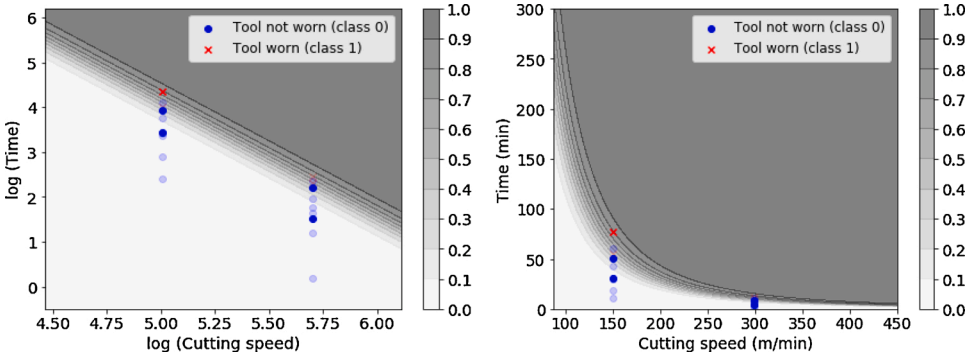

Fig. 9. Probability of tool worn as a function of cutting speed from the trained logistic model with logarithmic transformation of input variables; the train data points are shown in light and the test data points are shown in dark. The left panel shows the results in the logarithmic space and the right panel in the original space.

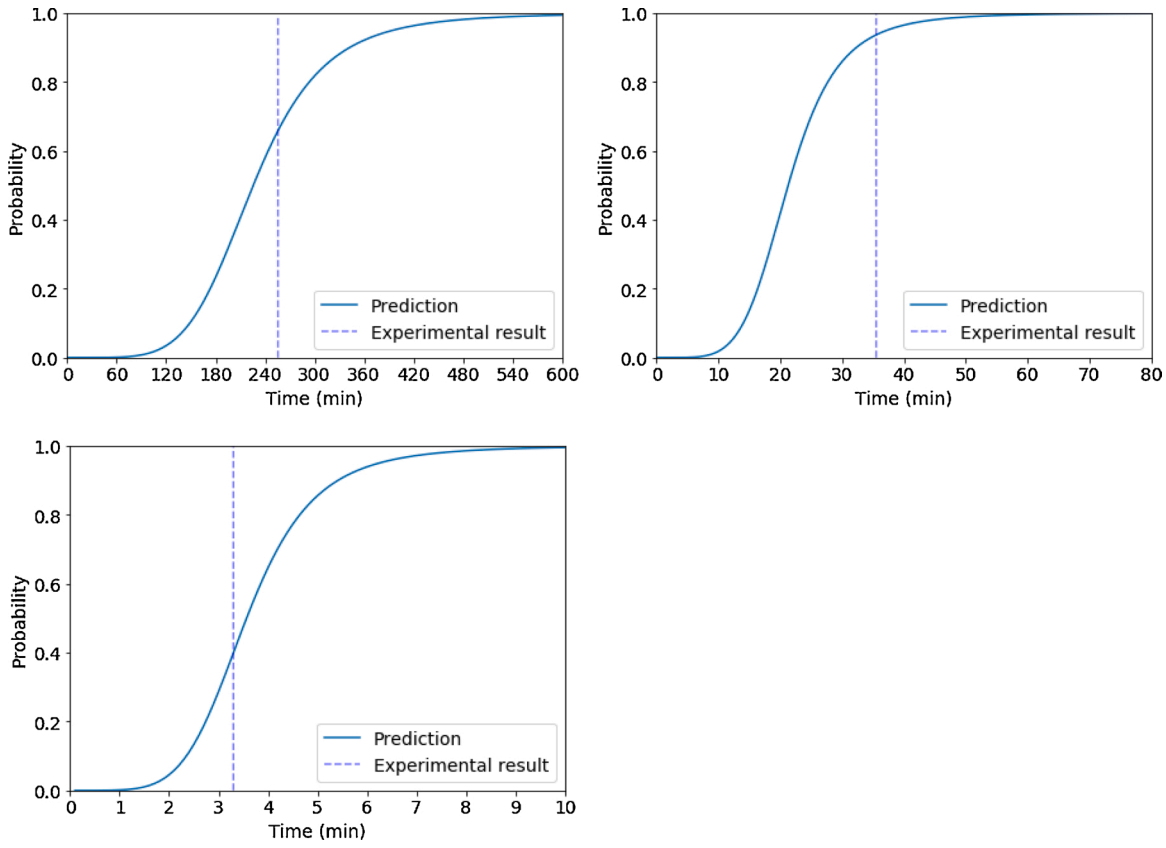

Fig. 10. Probability of tool worn (class 1) as a function of time and the experimental result at 89.8 m/min (1500 rpm) (top left panel), 224.4 m/min (3750 rpm) (top right panel), and 448.8 m/min (7500 rpm) (bottom left panel). Note that the scale in the horizontal axis for each cutting speed is different.

Comparing Eq. 9 and Eq. 11, the logistic classification model parameters may be directly related to the empirical Taylor tool life coefficients as shown in Eq. 12. This provides the desired interpretability for the trained logistic classification model.

$$n = \frac{\theta_2}{\theta_1}\,; C = e^{-n\frac{\theta_0}{\theta_2}} \tag{12}$$

To illustrate the approach, the trained logistic model parameters using the logarithmic transformation of the input data were $\theta_0 = -89.57$, $\theta_1 = 13.57$, and $\theta_2 = 5.26$. The logistic model parameters were converted into the equivalent Taylor tool life coefficients using Eq. 12 to obtain $n = 0.388$, and $C = 735.2$ m/min. A Taylor tool life model fit to the Table 1 data results in $C = 698.8$ m/min and $n = 0.372$. Fig. 11 shows a comparison between the Taylor tool life model fit and the logistic classification decision boundary. It is seen that the logistic classification decision boundary shows good agreement with the Taylor tool life equation. Note that the logistic classifier input was obtained from binary data for tool worn (class 0) or tool worn (class 1); the time for the end of tool life was not provided.

To evaluate the influence of the input data on the logistic model predictions, a Monte Carlo simulation was performed. The procedure to generate ten data points at each cutting speed, as described in Section 2, was repeated 100 times. For each dataset, the logistic model was trained using a logarithmic transformation of the inputs. Fig. 12 shows the 100 decision boundaries from the Monte Carlo simulation. The Taylor tool life fit is also included for reference. As expected, the decision boundary depends on the underlying dataset. If the underlying dataset has large

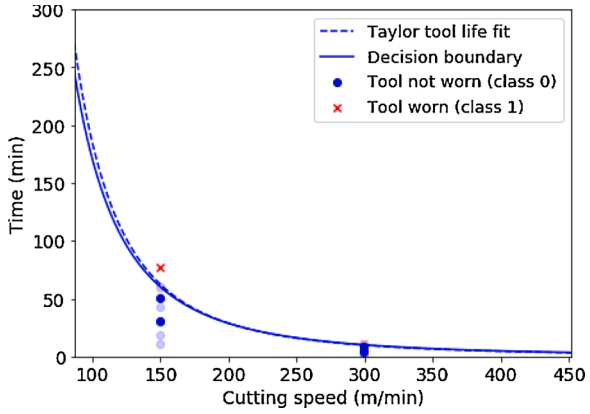

Fig. 11. Comparison between the Taylor tool life fit and logistic classifier decision boundary with logarithmic transformation of the inputs; results show good agreement.

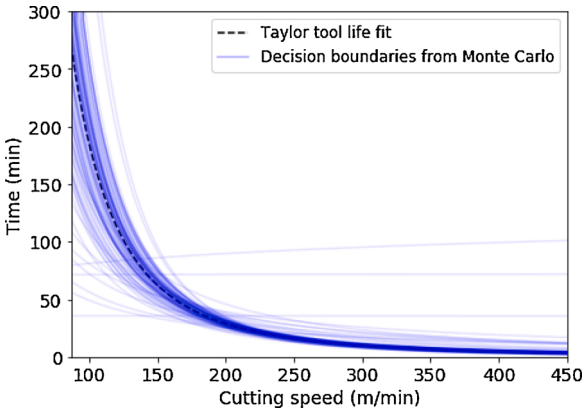

Fig. 12. Decision boundaries from the Monte-Carlo simulation.

uncertainties where the tool worn (class 0) and tool not worn (class 1) data points overlap, or if a cutting speed does not include tool worn (class 1) data points, the decision boundary deviates substantially from the Taylor tool life model. To illustrate, in Fig. 12, three distinct decision boundaries predict a small increase in tool life as a function cutting speed. The probability of tool worn (class 1) and the associated dataset for one decision boundary is shown in Fig. 13. For the dataset shown in Fig. 13, there were no tool worn (class 1) data points recorded at 299.2 m/min. As a result, the logistic model with the logarithmic transformation cannot capture the non-linear exponential decrease in tool life with cutting speed. The equivalent Taylor tool life coefficients calculated using Eq. 12 were n = -6.5, and C = 0 m/min. The equivalent Taylor tool life coefficients can quickly indicate the quality of the trained model. For example, it is known that the coefficient n in the Taylor tool life equation is typically between 0.1 and 0.4 [1]. Therefore, a trained model where the value of n is negative or large indicates that there is insufficient data. Similarly, the value C is the cutting speed to obtain a tool life of 1 min. An extremely large value indicates that the model is overpredicting tool life. In such cases, additional data can be collected to retrain the model. This can be done by running the tools at some cutting speeds to the end of life to get the tool worn data points. Note that for the given dataset, the trained model still accurately predicts the test data points. However, as shown, an examination of the Taylor tool life model coefficients calculated from the trained logistic model can indicate the quality of the trained model.

There have been recent studies in the literature for physics-guided neural networks. This is achieved either by modifying the loss function or adding constraints on the weights and the bias terms for the NN [34,35]. The proposed physics-guided logistic model with logarithmic transformation of the inputs has the following advantages over a physics-guided NN. First, the power law relationship between tool life and cutting speed can be explicitly modeled with the logarithmic transformation of the inputs. Second, as shown in Eq. 12, the logistic model coefficients can be converted to the Taylor tool life constants. The Taylor tool life constants calculated from the logistic model can be used to quickly validate the model results. Third, the Taylor tool life constants can be archived for different tool-material combinations. Finally, the model is simple to implement using in-built ML libraries (such as scikit-learn for Python [28,29]).

5. Process parameter optimization for minimizing cost

This section describes a probabilistic cost equation for pre-process optimization of machining parameters using the physics guided logistic model tool life predictions. It does not currently include the {spindle speed, depth of cut} constraints imposed by unstable cutting conditions (or chatter) [36] but can be expanded in future work. As noted in Section 1.0, there exists a machining cost trade-off between material removal rate and tool life. The machining cost per part is shown in Eq. 13 [1], where C_p is the machining cost per part in \$, r_m is machine tool rate in \$/min, t_m is the machining time in minutes, t_{ch} is the time required to change the tool in minutes, C_{te} is the cost per tool edge in \$, and T is the tool life in minutes.

$$C_p = t_m r_m + \frac{(t_{ch} r_m + C_{te}) t_m}{T} \tag{13}$$

The deterministic cost per part equation is modified in two ways. First, as shown from the results in Table 1, tool life is non-repeatable due to tool-to-tool performance variation. Therefore, the non-repeatability of tool life needs to be included in the machining cost per part equation. This is achieved by considering tool life as a probability distribution. The probability distribution in tool life can then be propagated to the machining cost per part. The optimal process parameters minimize the expected cost calculated using the machining cost per part distribution. Second, in a production environment, it is often infeasible to complete a fractional number of parts per tool edge or tool edge per part because it requires changing the tool in the middle of a cut. An integer number of parts per tool edge or tool edges per part also ensures that the tool change location remains constant and does not change for every part. For example, if the tool life is 12 min and the machining time per part is 10 min, it is convenient to use a new tool edge per part. In this case, the part per insert edge would be one. To address this, a new term called parts per tool edge is defined. If the tool life is greater than the cut time, the parts per tool edge is the maximum integer number of parts that can be machined with a single tool edge. If the tool life is less than the machining time per part, the parts per tool edge is the reciprocal of the minimum integer number of tools required to machine the part. The modified cost equation per part is shown in Eq. 14, where p_{te} is the number of parts per tool edge.

$$C_p = t_m r_m + \frac{(t_{ch} r_m + C_{te})}{p_{te}} \tag{14}$$

The parts per tool edge can be calculated using Eq. 15, where $\lfloor \frac{T}{t_m} \rfloor$ denotes 'floor' of the ratio of tool life to cutting time and $\lceil \frac{t_m}{T} \rceil$ denotes the 'ceiling' of the ratio of machining time to tool life.

$$p_{te} = \begin{cases} \lfloor \frac{T}{t_m} \rfloor & \text{if } T \geq t_m \\ \frac{1}{\lceil \frac{t_m}{T} \rceil} & \text{if } T < t_m \end{cases} \tag{15}$$

To illustrate, if the tool life is 12 min and machining time is 10 min, $p_{te} = 1$. If the tool life is 8 min and the machining time is 10 min, $p_{te} = 0.5$ (corresponding to two tool edges per part). The probability distribution of tool life is taken from the prediction of the trained logistic classifier. At a selected cutting speed, the probability distribution in tool life is first propagated to parts per tool edge using Eq. 15. This results in a discrete distribution for parts per edge. The parts per edge distribution is then propagated to machining cost per part using Eq. 14. The expected cost per part is calculated as the sum of the product of the discrete cost per part values and the associated probabilities.

To illustrate the method, consider a part with a volume to be removed of 1×10^5 mm^3. As a simplification, the machining time at each spindle speed may be calculated using the (mean) material removal rate and the volume to be removed. The cost per tool edge was taken as \$2.50, the tool change time was 2 min, and the machine tool rate was \$2/min. Fig. 14 shows the expected cost as a function of cutting speed; the optimal spindle speed to minimize the expected cost is 374 m/min (6250 rpm). Fig. 15 shows the tool life distribution (top left panel), the

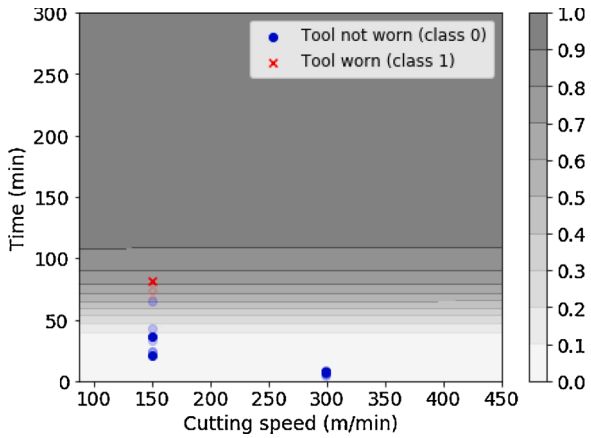

Fig. 13. Probability of tool worn (class 1) for the outlier predictions; no tool worn (class 1) data is available at 299.2 m/min so the model accuracy is compromised.

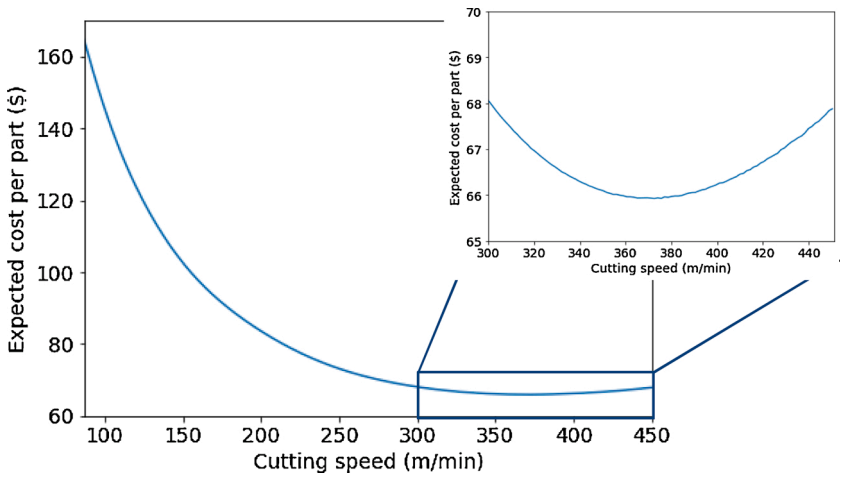

Fig. 14. Expected cost per part as a function of cutting speed; the minimum expected cost is $65.90 at 374 m/min (6250 rpm).

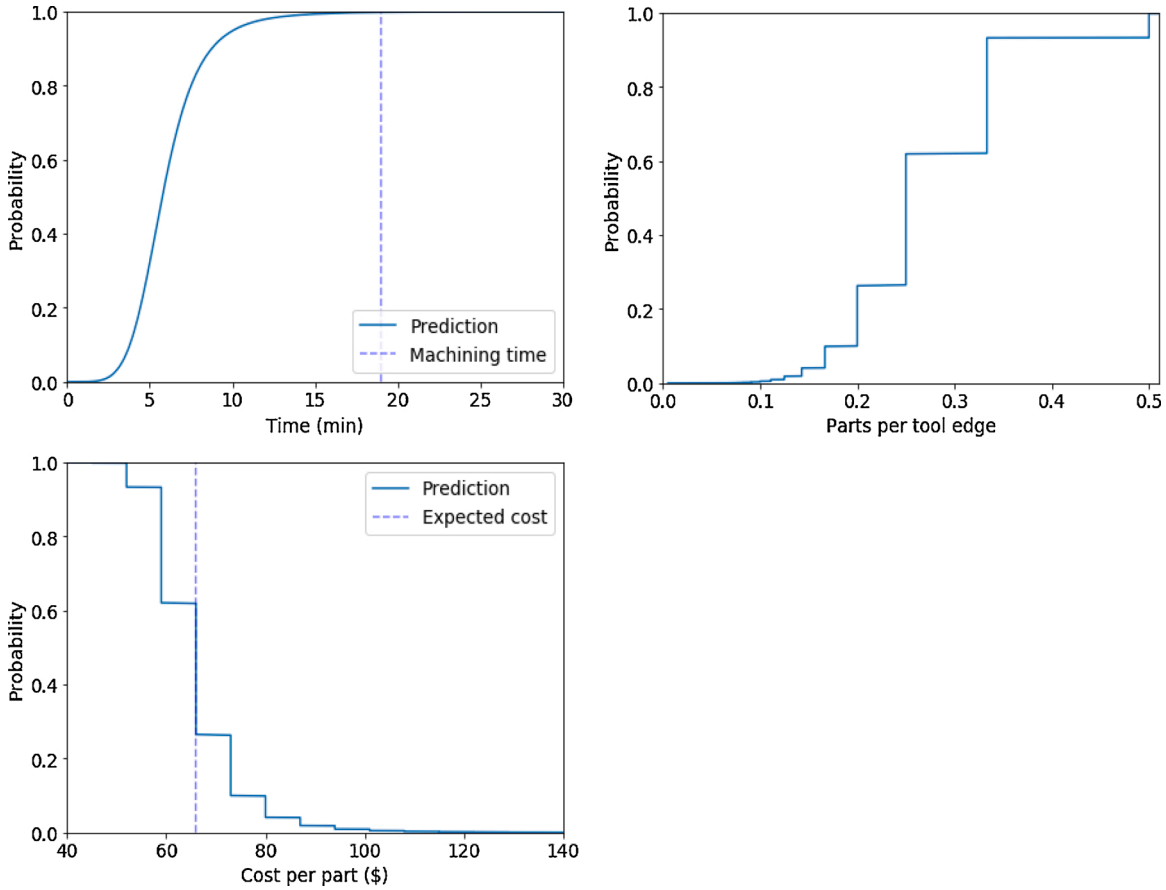

Fig. 15. Probability of worn tool at 374 m/min; the top left panel shows the probability of tool worn from the logistic model, the top right panel shows the distribution of parts per tool edge, and the bottom left panel shows the distribution of machining cost per part.

parts per edge distribution (top right panel), and machining cost per part distribution (bottom left panel) at 374 m/min. The different distributions shown in Fig. 15 can be explained as follows. As noted, the probability of a tool worn classification from the trained logistic classifier is the probability that tool life would be less than or equal to the values on the horizontal axis. The part per tool edge distribution gives the probability that the number of parts per tool edge would be less than or equal to the value on the horizontal axis. The part per tool edge distribution is calculated using Eq. 15 for different tool life values, sampled from the tool life distribution. Note that the part per tool edge distribution is discrete instead of continuous; this is because the parts per tool edge is an integer if tool life is greater than the machining time or a reciprocal of an integer if the tool life is less than the machining time. This is shown in Eq. 15. Since the tool life prediction is less than the machining time at 374 m/min, more than one tool edge would be required to machine the part. The most likely outcome is four tool edges per part (or 0.25 parts per tool edge); this is seen from the distribution of parts per tool edge shown in Fig. 15 (top right panel). Each value of part per tool edge corresponds to a machining cost per part which results in a probability distribution for machining cost per part. Since the machining cost per

part increases with a reduction in the number of parts per tool edge at a given cutting speed, the distribution of machining cost per part gives the probability that machining cost would be greater than or equal to the value on the horizontal axis (shown in the Fig. 15 bottom left panel). The expected cost from the cost per part distribution shown in Fig. 15 (bottom left panel) is $65.90. Note that the actual cost at 374 m/min will depend on the actual number of parts per tool edge, which, in turn, depends on the observed tool life and machining time. In a production environment, the optimal cutting speed can be validated as follows. Tool checks can be added at regular intervals in a toolpath based on the parts per tool edge distribution. If the tool wear is near or exceeds the threshold limit, the tool can be replaced and the parts per tool edge noted for subsequent parts. This reduces the risk of validating the optimal cutting speed by preventing damage to the part due to excessive tool wear or catastrophic tool breakage.

To confirm the optimal cutting speed, a tool life experiment was performed at 374 m/min. The *FWW* was measured as 0.21 mm at 6.9 min and 0.37 mm at 10.0 min. The tool life for *FWW* to reach 0.3 mm was calculated by linear interpolation between the two measurements as 8.6 min. Based on a machining time of 18.9 min, the parts per tool edge is 0.33, giving a cost of $58.90 (calculated using Eq. 14. and Eq. 15). Note that the tool life needs to be between 6.3 minutes–9.5 minutes to enable 0.33 parts per edge.

6. Discussion

In this section, a method for data augmentation for imbalanced datasets, a hybrid model that combines the linear and logarithmic input, and a method for extending the model to multiple variables are presented.

6.1. Data augmentation

As shown in Section 4, the logistic model predictions depend on the underlying data. Fig. 13 showed a dataset where zero tool worn (class 1) data points were recorded at 299.2 m/min. As a result, the logistic model with the logarithmic input transformation could not model the non-linear reduction in tool life with cutting speed. In such a scenario, additional data may be collected to retrain the model. However, in a production environment, additional data at different tool replacement times or different cutting speeds may not be feasible. In this case, synthetic tool worn (class 1) data points can be added based on user experiences. The user assigns a tool life value based on experience; the objective is to add synthetic tool worn data points where the tool will be worn with certainty based on user experience. A tool worn (class 1) data point can be added at the user-assigned value. Alternatively, the maximum measured wear value from the tool not worn (class 0) data points can be linearly extrapolated to the threshold wear limit to obtain the desired tool worn (class 1) data point. To illustrate the approach, consider a case where the user assigns a tool life of 15 min at 299.2 m/min. The value of 15 min is decided by the user as the value where the tool will be worn (maximum *FWW* exceeds 0.3 mm) with certainty. Fig. 16 shows the probability of tool worn from the logistic classifier with two synthetic tool worn data points added at {15 min, 299.2 m/min} to the training dataset. In Fig. 16, the right panel shows the magnified view of the data at 299.2 m/min, where the synthetic tool worn data points are observed at 15 min. Note that the mean tool life from the three tests at 299.2 m/min (listed in Table 1) was 9.8 min. As seen from Fig. 16, the logistic model can describe the relationship using the augmented dataset. Recall that the tool life is the logistic classifier decision boundary that separates the two classes: tool worn (class 0) and tool not worn (class 1). Adding synthetic tool worn data at 299.2 m/min enables the classifier to model the boundary that separates the two classes. The logarithmic transformation of the inputs enables the logistic classifier to model the exponential decrease in tool life with cutting speed. To evaluate the influence of the synthetic data on the trained logistic model, the logistic model was trained separately with 10 min and 20 min data points added to the training dataset. Fig. 17 shows the probability of tool worn with synthetic data added at 10 min (left panel) and 20 min (right panel).

As seen from Figs. 16 and 17, the probability of tool worn distribution is wider for the augmented dataset with the 15 min and 20 min data as compared to 10 min augmented data. For the 10 min augmented dataset, there is a separation of 6 min between the two classes at 149.6 m/min and 1 min at 299.2 m/min. As a result, the model finds the boundary separating the two classes, but the probability distribution of tool worn (class 1) is narrow. The separation between the two classes increases with the 15 min and 20 min datasets resulting in a wider distribution. Therefore, as stated, synthetic tool worn data should be added where the tool wear will exceed the threshold limit with certainty. This is important since the predicted tool life from the logistic classifier separates the two classes. As stated, the objective of the tool life prediction model is pre-process optimization of process parameters. If the 'true' tool life lies in the predicted probability distribution, checks can be added when testing at the optimum cutting speed to reduce the risk in testing.

6.2. Hybrid logistic model

As shown in Section 4, the non-linear relationship between tool life and cutting speed defined by the Taylor tool life equation can be modeled using a logarithmic transformation of the inputs. The logistic model parameters can then be directly correlated to the Taylor tool life coefficients. However, there may be certain tool and material combinations where tool life does not follow the exponential decrease with cutting speed as defined by the Taylor tool life equation. For example, ceramic tools may not show the exponential decrease in tool life with cutting speeds [37,38]. In such cases, the logistic model with

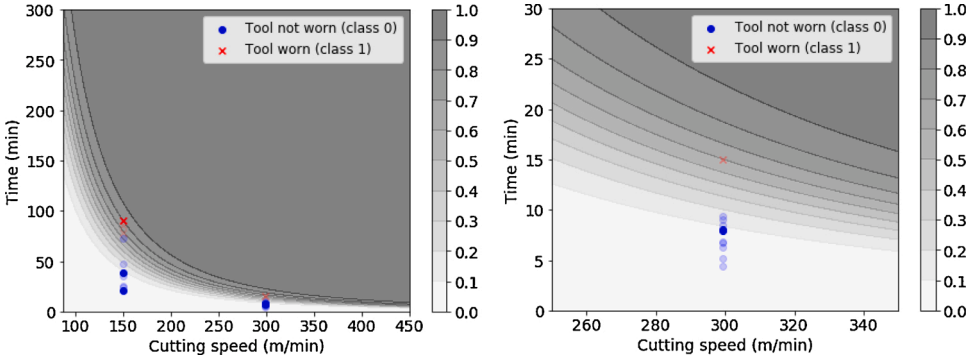

Fig. 16. Probability of tool worn with the trained logistic model with logarithmic transformation of input variables with 10 data points at 149.6 m/min and 299.2 m/min each; two tool worn (class 1) data points were added to the training data for 299.2 m/min (shown in the right panel).

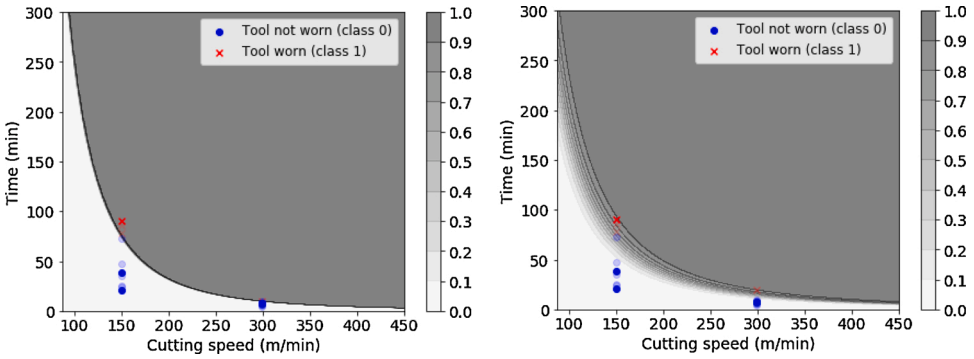

Fig. 17. Probability of tool worn with the log-transformed logistic classifier with synthetic data added at 299.2 m/min; 10 min (left panel) and 20 min (right panel).

logarithmic transformation may overpredict tool life when extrapolating outside the input data range. To make the logistic model generalizable to different tool life responses as a function of cutting speed, a hybrid model consisting of the original input values in addition to the log-transformed inputs was evaluated. For a one-dimensional case with cutting speed and time as inputs, the decision boundary equation for the hybrid model is given by Eq. 16.

$$\theta_0 + \theta_1(V) + \theta_2(T) + \theta_3\log(V) + \theta_4\log(T) = 0 \tag{16}$$

The idea behind the hybrid model is that if the underlying tool life follows the power law given by the Taylor tool life equation, θ_1 and θ_2 will be zero giving the log-transformed equation shown in Eq. 10. However, if tool life decreases linearly with cutting speed in the selected range, θ_3 and θ_4 will be zero giving the original logistic model decision boundary equation shown in Eq. 6. If the tool life follows a curve that cannot be fully defined with a linear or the Taylor tool life equation, the logistic model coefficients will be comparable. To evaluate the hybrid logistic model, it was trained using the data shown in Fig. 4. The procedure to train the logistic model is described in Section 4. Note that the input variables need to be normalized due to the magnitude difference between the input (cutting speed and time) and the logarithm values of the inputs. The input values were normalized between 0 and 1 based on the minimum and maximum values for cutting speed and time shown in Fig. 9. Fig. 18 shows the probability of tool worn (left panel) and the decision boundary (right panel) for the hybrid model. The decision boundary for the logistic model with logarithmic input transformation is also shown for reference (see the right panel in Fig. 18). As seen from Fig. 18, the hybrid model does not predict an exponential increase in tool life at cutting speeds less than 149.6 m/min. Fig. 19 shows the tool life distribution for the hybrid model at 149.6 m/min and 448.8 m/min. The prediction from the logistic model with logarithmic transformation is also shown for comparison. As seen from Figs. 18 and 19, the hybrid model underpredicts tool life at 89.8 m/min (1500 rpm) and overpredicts at 448.8 m/min (7500 rpm). This is because, with the limited dataset, the hybrid model fits a combination of the linear and an exponential model. Therefore, tool life does not increase or decrease exponentially when extrapolating the results outside the training dataset. Note that the results shown in Fig. 19 do not show a clear benefit for the hybrid model over the logistic with the logarithmic transformation model; however, differences in the model predictions are observed.

The logistic model coefficients for the hybrid model were calculated as $\theta_0 = -79.2$, $\theta_1 = 17.7$, $\theta_2 = -1.43$, $\theta_3 = 18.2$, and $\theta_4 = 56.6$. Although a direct comparison to the Taylor tool life model coefficients cannot be made with the hybrid model, assessments on the nature of the logistic model fit can be made based on the magnitude of the coefficients. As noted, if $\theta_3 \gg \theta_1$ and $\theta_4 \gg \theta_2$, the underlying tool life curve follows the Taylor tool life equation. If the opposite is true, the tool life relationship is linear. If the magnitudes are similar, the tool life curve is a combination of the Taylor tool life and linear equation. Under a limited dataset, the hybrid model is a combination of the linear and the exponential Taylor tool life models. The hybrid model offers flexibility in modeling tool life curves that do not follow the exponential Taylor tool life equations. In practical applications, both the logistic model with logarithmic transformation of the inputs and the hybrid model with linear and logarithmic-transformed inputs may be evaluated and the model which performs better may be chosen to optimize machining parameters.

6.3. Extension to multiple dimensions

The physics-guided logistic method presented in Section 4 can be extended to include the effects of feed per revolution using the extended Taylor type tool life equation. The new Taylor tool-type tool life equation is given by [1]:

$$V^p F^q T = C \tag{17}$$

where p, q, and C are Taylor tool life coefficients and F is the feed per

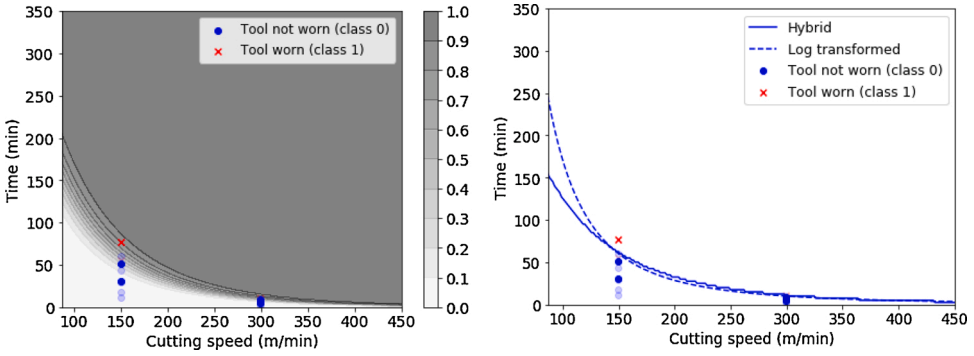

Fig. 18. Probability of tool worn (left) and the decision boundary (right) with cutting speed from the trained hybrid logistic model; the train data points are shown in light and the test data points are shown in dark. The decision boundary from the log-transformed logistic model is also shown for comparison.

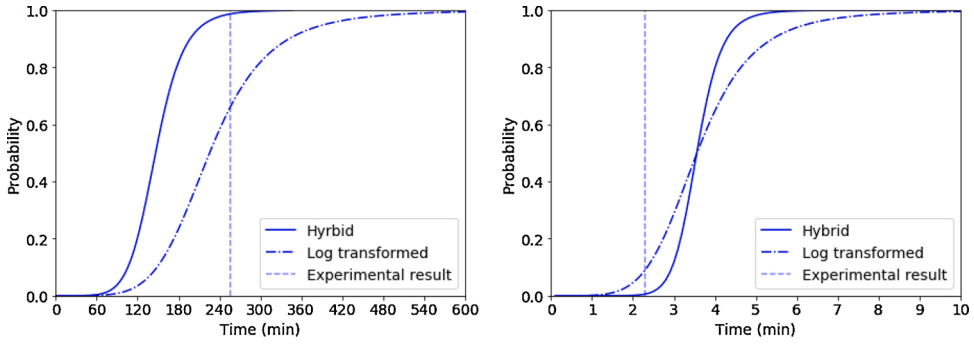

Fig. 19. Comparison between hybrid and log-transformed logistic model at 89.9 m/min (left) and 448.8 m/min (right).

revolution. Taking the logarithm and rearranging gives:

$$\log(T) = log(C) - plog(V) - qlog(F) \tag{18}$$

For the three inputs of cutting speed, feed per revolution, and time, the decision boundary for the logistic classification with logarithmic transformation of the input features is given by:

$$\theta_0 + \theta_1 \log(V) + \theta_2 \log(F) + \theta_3 \log(T) = 0 \tag{19}$$

$$\log(T) = -\frac{\theta_0}{\theta_3} - \frac{\theta_1}{\theta_3}\log(V) - \frac{\theta_2}{\theta_3}\log(F) \tag{20}$$

Comparing Eq. 18 and Eq. 20, the logistic classification model parameters may be directly related to the Taylor-type tool life coefficients.

$$C = -\frac{\theta_0}{\theta_3}; p = \frac{\theta_1}{\theta_3}; q = \frac{\theta_2}{\theta_3} \tag{21}$$

As described in Section 4.0, the logistic classification model can be trained with a logarithmic transformation of the inputs, cutting speed, feed per revolution, and time. As noted, the value of the Taylor-type coefficients determined from the logistic model can be used to quickly validate the model. The value of p is typically between 2 and 4 and q is typically between 1 and 3 [1]. Furthermore, the hybrid model described in Section 6.2 can be used to extend the tool life logistic model to include additional variables such as cutting fluid (type and the application method) and the tool geometry. In this case, a logarithmic transformation for cutting speed, feed rate, and time can be performed while cutting fluid and tool geometry inputs are included without any transformation. The hybrid model offers the required flexibility to include multiple variables in the tool life model.

7. Conclusions

This paper presented a physics-guided logistic classifier for tool life modeling in production environments. The non-linear relationship between tool life and cutting speed was incorporated in the linear logistic classifier by performing a logarithmic transformation of the inputs (cutting speed and time). The logistic classifier is therefore linear in the logarithmic space and follows the non-linear power-law relationship given by the Taylor tool life equation in the original space. A method to extract the Taylor tool life coefficients from the trained logistic model was presented by comparing the logistic model decision boundary to the Taylor tool life equation. Validation results in milling showed that the logistic classifier with a logarithmic transformation of the inputs can predict tool life with binary data as inputs. The logistic model predictions were used to optimize process parameters using a modified probabilistic cost equation. The distribution in tool life was propagated to the machining cost equation through a parts per tool edge term. The proposed method offers a practical and robust approach for tool life modeling and machining parameter optimization in a production environment. Future work will focus on adding additional variables, such as tool geometry and coolant/lubricant use as inputs, and testing and validating the algorithm using actual data from a production environment.

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.

Acknowledgments

The authors would like to thank Dr. Luke Scime, Oak Ridge National Laboratory, for numerous helpful comments and feedback which helped improve the quality of the manuscript.

This research was supported by the DOE Office of Energy Efficiency and Renewable Energy (EERE), Manufacturing Science Division, and used resources at the Manufacturing Demonstration Facility, a DOE-EERE User Facility at Oak Ridge National Laboratory.

References

[1] Tlusty J. Manufacturing process and equipment. NJ: Prentice Hall, Upper Saddle River; 2000. pp. 463.
[2] Olortegui-Yume JA, Kwon PY. Tool wear mechanisms in machining. Int J Mach Machinability of Matl 2007;2(3–4):316–34.
[3] Özel T, Karpat Y. Predictive modeling of surface roughness and tool wear in hard turning using regression and neural networks. Int J Mach Tools and Manuf 2005;45 (4–5):467–79.
[4] Quiza R, Figueira L, Davim JP. Comparing statistical models and artificial neural networks on predicting the tool wear in hard machining D2 AISI steel. Int J Adv Manuf Technol 2008;37(7–8):641–8.
[5] Alauddin M, El Baradie MA, Hashmi MS. Prediction of tool life in end milling by response surface methodology. J Mater Process Technol 1997;71(3):456–65.
[6] Takeyama H, Murata R. Basic investigation of tool wear. ASME J Ind Eng 1963;85 (33):C38.
[7] Usui E, Shirakashi T, Kitagawa T. Analytical prediction of cutting tool wear. Wear 1984;100(1-3):129–51.
[8] Usui E, Shirakashi T, Kitagawa T. Analytical prediction of three dimensional cutting process – part 3: cutting temperature and crater wear of carbide tool. J Inst Eng 1978;100(2):236–43.
[9] Xie LJ, Schmidt J, Schmidt C, Biesinger F. 2D FEM estimate of tool wear in turning operation. Wear 2005;258(10):1479–90.
[10] Halila F, Czarnota C, Nouari M. New stochastic wear law to predict the abrasive flank wear and tool life in machining process. Proc Inst Mech. Eng Part J J Eng Tribol 2014;228(11):1243–51.
[11] Salonitis K, Kolios A. Reliability assessment of cutting tool life based on surrogate approximation methods. Int J Adv Manuf Tech 2014;71(5–8):1197–208.
[12] Salonitis K, Kolios A. Reliability assessment of cutting tools life based on advanced approximation methods. Proc CIRP 2013;8:397–402.
[13] Karandikar JM, Abbas AE, Schmitz TL. Tool life prediction using Bayesian updating. Part 1: milling tool life model using a discrete grid method. Precis Eng 2014;38(1):9–17.
[14] Karandikar JM, Abbas AE, Schmitz TL. Tool life prediction using Bayesian updating. Part 2: turning tool life using a Markov Chain Monte Carlo approach. Precis Eng 2014;38(1):18–27.
[15] Karandikar JM, Schmitz TL, Abbas AE. Spindle speed selection for tool life testing using Bayesian inference. Int J Ind Manuf Syst Eng 2012;31(4):403–11.
[16] Samuel AL. Some studies in machine learning using the game of checkers. IBM J Res And Dev 1959;3(3):210–29.

[17] Mitchell TM. Machine learning. 1997. Burr Ridge, IL: McGraw Hill; 1997. p. 870–7. 45:37.
[18] Wang Y, Zheng L, Wang Y. Event-driven tool condition monitoring methodology considering tool life prediction based on industrial internet. J Manuf Sys 2021;58 (January):205–22. 1.
[19] Liu R, Kothuru A, Zhang S. Calibration-based tool condition monitoring for repetitive machining operations. J ManufSys 2020;54(January):285–93. 1.
[20] Wang J, Li Y, Zhao R, Gao RX. Physics guided neural network for machining tool wear prediction. J Manuf Sys 2020;57:298–310.
[21] Corne R, Nath C, El Mansori M, Kurfess T. Study of spindle power data with neural network for predicting real-time tool wear/breakage during inconel drilling. J Manuf Sys 2017;43:287–95.
[22] Xie Y, Lian K, Liu Q, Zhang C, Liu H. Digital twin for cutting tool: modeling, application and service strategy. J Manuf Sys 2021;58:305–12.
[23] Bhuiyan MS, Choudhury IA, Dahari M. Monitoring the tool wear, surface roughness and chip formation occurrences using multiple sensors in turning. J Manuf Sys 2014;33(4):476–87.
[24] Wang J, Wang P, Gao RX. Enhanced particle filter for tool wear prediction. J Manuf Sys 2015 Jul;1(36):35–45.
[25] Karandikar J. Machine learning classification for tool life modeling using production shop-floor tool wear data. Proc. Manuf. 2019;34:446–54.
[26] International standards organisation, ISO 8688-2: tool life testing in milling-part 2 : end milling. 1989.
[27] Shalev-Shwartz S, Ben-David S. Understanding machine learning: from theory to algorithms. Cambridge University Press; 2014. May 19.
[28] Géron A. Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: concepts, tools, and techniques to build intelligent systems. O'Reilly Media; 2019. Sep 5.
[29] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine learning in Python. J Mach Learn Res 2011;12(October):2825–30.
[30] Buitinck L, Louppe G, Blondel M, Pedregosa F, Mueller A, Grisel O, et al. API design for machine learning software: experiences from the scikit-learn project. arXiv preprint arXiv 2013;1309:0238.
[31] Taylor FW. On the art of cutting metals. Am Soc Mech Eng 1906.
[32] Johansson D, Hägglund S, Bushlya V, Ståhl JE. Assessment of commonly used tool life models in metal cutting. Proc Manuf 2017;11:602–9.
[33] Li B. A review of tool wear estimation using theoretical analysis and numerical simulation technologies. Int J Refractory Metals Hard Matl 2012;35:143–51.
[34] Karpatne A, Watkins W, Read J, Kumar V. Physics-guided neural networks (pgnn): an application in lake temperature modeling. arXiv preprint arXiv 2017;1710: 11431.
[35] Wang J, Li Y, Zhao R, Gao RX. Physics guided neural network for machining tool wear prediction. J Manuf Syst 2020;57:298–310.
[36] Schmitz T, Smith KS. Machining dynamics: frequency response to improved productivity. 2nd ed. New York, NY: Springer; 2019.
[37] Altin A, Nalbant M, Taskesen A. The effects of cutting speed on tool wear and tool life when machining Inconel 718 with ceramic tools. Mater Des 2007;28(9): 2518–22.
[38] Astakhov VP, Davim JP. Tools (geometry and material) and tool wear. In machining. London: Springer; 2008. p. 29–57.