

**MACHINE LEARNING-BASED MODELING AND MONITORING OF  
MACHINING PROCESSES AND TOOL WEAR**

by

**ARASH EBRAHIMI ARAGHIZAD**

Submitted to the Graduate School of Engineering and Natural Sciences

in partial fulfillment of

the requirements for the degree of Doctor of Philosophy

Sabancı University

July 2024

**MACHINE LEARNING-BASED MODELING AND MONITORING OF  
MACHINING PROCESSES AND TOOL WEAR**

Arash Ebrahimi Araghizad 2024 ©

All Rights Reserved

## **ABSTRACT**

### **MACHINE LEARNING-BASED MODELING AND MONITORING OF MACHINING PROCESSES AND TOOL WEAR**

Arash Ebrahimi Araghizad

Manufacturing Engineering Ph.D. Dissertation, July 2024

Dissertation Supervisor: Prof. Dr. Erhan Budak

Dissertation Co-Advisor: Assoc. Prof. Kemal Kiliç

Milling processes are a basis of manufacturing across a variety of industries, including aerospace, automotive, and heavy machinery. Effective monitoring of these processes is essential to ensuring high-quality production, minimizing downtime, and extending tool life. Inefficient monitoring can lead to defects in machined parts, excessive tool wear, and ultimately, significant financial and material waste. Traditionally, the monitoring of milling operations has relied heavily on direct human supervision, periodic testing, and sensor-based monitoring systems, which are based on time-consuming teach cycles. These existing monitoring systems, mainly targeted at tool condition monitoring, cannot accurately identify the source of variation or fault, making them both error-prone and resource-intensive. This thesis seeks to revolutionize this traditional approach by implementing advanced machine learning (ML) techniques integrated with physics-based simulations, significantly reducing the reliance on extensive physical testing and manual oversight.

The enhanced monitoring capabilities offered by these advanced technologies enable precise control over the milling process, ensuring optimal tool engagement and machining parameters. This leads to improved consistency in product quality and a substantial reduction in waste, which is crucial for maintaining competitiveness in fast-paced

markets. Moreover, intelligent monitoring systems can predict tool wear and potential failures before they occur, allowing for preemptive maintenance and scheduling. This not only extends the lifespan of milling equipment but also ensures continuous production without unexpected interruptions, thereby enhancing overall manufacturing efficiency.

The first section of the thesis presents a series of innovative hybrid models, known as physics-informed machine learning (PIML), which excel in predicting milling forces, tool wear, and tool-related faults. By combining limited experimental data with detailed simulation outputs, these models achieve predictive accuracies up to 98%. Demonstrated across various materials and tool configurations, the models' adaptability and scalability underscore their potential for widespread industrial application.

Subsequent sections elaborate on the development of an advanced fault detection system, designed specifically for real-time applications in unmanned manufacturing environments. This system, employing refined force models and sophisticated machine learning algorithms, not only detects deviations with over 96% accuracy but also pinpoints the source of these faults. By accurately identifying not just the occurrence of anomalies but also their origins, the system enables targeted interventions, thereby optimizing manufacturing processes and significantly reducing operational costs. This dual capability of detection and precise source identification enhances the system's effectiveness in maintaining continuous production flow and minimizing downtime.

Additionally, the thesis explores tool wear prediction using hybrid modeling approaches that integrate mechanistic insights with diverse ML algorithms. This approach significantly reduces the need for extensive wear testing, facilitating a more streamlined and economically viable monitoring process.

In conclusion, the research presented in this thesis not only advances the field of intelligent manufacturing monitoring by providing robust predictive tools but also establishes a solid foundation for future enhancements. By transforming traditional monitoring methods into more intelligent, efficient, and adaptive systems, this work pioneers a new era of manufacturing that aligns with the demands of modern industry.

## ÖZET

### İŞLEME PROSESLERİ VE TAKIM AŞINMASININ MAKİNE ÖĞRENMESİ TABANLI MODELLENMESİ VE İZLENMESİ

Arash Ebrahimi Araghizad

ÜRETİM MÜHENDİSLİĞİ DOKTORA TEZİ, TEMMUZ 2024

Tez Danışmanı: Prof. Dr. ERHAN BUDAK

Doktora Tezi Danışman Yardımcısı: Doç. Dr. Kemal Kılıç

Talaşlı imalat işlemleri, havacılık, otomotiv ve ağır makine endüstrileri dahil birçok sektörde imalatın temelini oluşturur. Bu işlemlerin etkin bir şekilde izlenmesi, yüksek kaliteli üretimin sağlanması, iş durma sürelerinin en aza indirilmesi ve takım ömrünün uzatılması açısından hayati öneme sahiptir. Etkin olmayan izleme, işlenmiş parçalarda hatalara, aşırı takım aşınmasına ve sonuçta önemli mali ve malzeme israfına yol açabilir. Geleneksel olarak, frezeleme operasyonlarının izlenmesi büyük ölçüde doğrudan insan gözetimi, periyodik testler ve zaman alıcı öğretim döngülerine dayanan sensör tabanlı izleme sistemlerine dayanmaktadır. Bu mevcut izleme sistemleri, esas olarak takım durumunu izlemeye yönelik olup, varyasyonun veya arızanın kaynağını doğru bir şekilde belirleyemezler, bu da onları hem hata yapmaya yatkın hem de kaynak yoğun hale getirir. Bu tez, fizik tabanlı simülasyonlarla entegre edilmiş ileri makine öğrenimi (ML) tekniklerinin uygulanması suretiyle bu geleneksel yaklaşımı devrim niteliğinde değiştirmeyi amaçlamaktadır. Bu, kapsamlı fiziksel testlere ve manuel denetime olan bağımlılığı önemli ölçüde azaltır.

Bu ileri teknolojilerin sunduğu gelişmiş izleme yetenekleri, frezeleme işlemi üzerinde hassas kontrol sağlayarak optimal takım angajmanı ve işleme parametrelerini garanti eder. Bu, ürün kalitesindeki tutarlılığın artmasına ve atıkların önemli ölçüde azalmasına yol açar ki bu da hızlı hareket eden pazarlarda rekabetçiliği korumak için hayati önem taşır. Dahası, akıllı izleme sistemleri, oluşmadan önce takım aşınmasını ve olası arızaları

tahmin edebilir, bu da önleyici bakım ve planlama imkanı sunar. Bu, sadece frezeleme ekipmanlarının ömrünü uzatmakla kalmaz, aynı zamanda beklenmedik kesintiler olmaksızın sürekli üretimi de garanti eder, böylece genel üretim verimliliğini artırır.

Tezin ilk bölümü, frezeleme kuvvetlerini, takım aşınmasını ve takımla ilgili arızaları tahmin etmede üstün olan yenilikçi hibrit modeller serisini, fizik bilgilendirilmiş makine öğrenimi (PIML) olarak adlandırılan, sunar. Sınırlı deneysel veriler ile detaylı simülasyon çıktılarını birleştirerek, bu modeller %98'e varan tahmin doğruluklarına ulaşır. Çeşitli malzemeler ve takım konfigürasyonları üzerinde gösterilmiştir; modellerin uyum kabiliyeti ve ölçeklenebilirliği, geniş endüstriyel uygulama potansiyellerini vurgular.

Sonraki bölümler, insansız imalat ortamlarında gerçek zamanlı uygulamalar için özel olarak geliştirilen sofistike bir arıza tespit sisteminin geliştirilmesini detaylandırır. Gelişmiş kuvvet modelleri ve ML algoritmaları kullanarak, bu sistem %96'nın üzerinde bir doğrulukla sapmaları tanır, süreçleri optimize eder ve maliyetleri önemli ölçüde düşürür.

Ayrıca, tez çeşitli ML algoritmaları ile mekanistik içgörülerin entegre edildiği hibrit modelleme yaklaşımlarını kullanarak takım aşınmasının tahmin edilmesini araştırır. Bu yaklaşım, kapsamlı aşınma testlerine olan ihtiyacı önemli ölçüde azaltır, daha akıcı ve ekonomik olarak uygulanabilir bir izleme sürecini kolaylaştırır.

Sonuç olarak, bu tezde sunulan araştırmalar, sadece sağlam tahmin araçları sağlayarak akıllı imalat izleme alanını ilerletmekle kalmaz, aynı zamanda gelecekteki geliştirmeler için sağlam bir temel de oluşturur. Geleneksel izleme yöntemlerini daha akıllı, verimli ve uyumlu sistemlere dönüştürerek, bu çalışma modern endüstrinin talepleriyle uyumlu yeni bir üretim çağının öncüsü olur.

*To My Best Friend,  
My Wife,  
Naeimeh*

## TABLE OF CONTENTS

ABSTRACT.....	IV
ÖZET .....	VI
TABLE OF CONTENTS.....	IX
LIST OF TABLES.....	XII
LIST OF FIGURES .....	XIII
1. INTRODUCTION.....	1
1.1. Motivation.....	5
1.2. Objectives.....	6
1.3. Thesis Overview .....	8
2. LITERATURE REVIEW.....	10
2.1. Enhanced Milling Forces Estimation.....	11
2.2. Milling Process Parameter Identification.....	15
2.3. Tool Condition Monitoring (TCM).....	17
2.4. Tool Wear Monitoring .....	21
3. IMPROVING MILLING FORCE PREDICTIONS: A HYBRID APPROACH INTEGRATING PHYSICS-BASED SIMULATION AND MACHINE LEARNING FOR REMARKABLE ACCURACY ACROSS DIVERSE UNSEEN MATERIALS AND TOOL TYPES.....	25
3.1. Machine Learning Algorithms (ML) .....	25
3.1.1. Support Vector Regression (SVR).....	25
3.1.2. Least Square Gradient Boosting Algorithm (LSBoost).....	26
3.1.3. Random Forest (RF) .....	30
3.1.4. Hyperparameters Optimization.....	31

3.2. Hybrid Milling Force Prediction Methodology .....	33
3.2.1. Milling Force Modeling.....	34
3.2.2. Data Preparation .....	37
3.3. Results and Discussion.....	37
3.3.1. Cutting force prediction by hybrid ML model.....	37
3.3.2. Prediction accuracy of unseen material .....	46
3.3.3. Prediction accuracy of special milling tools .....	48
3.3.4. Models' performance for out-of-range input values.....	50
3.4. Summary .....	51
3.5. Future work .....	53
4. SMART TOOL-RELATED FAULTS MONITORING SYSTEM USING PROCESS SIMULATION-BASED MACHINE LEARNING ALGORITHMS .....	54
4.1. Milling Force Model .....	54
4.2. Applied Machine Learning for Fault Detection .....	55
4.2.1. Dataset for Training Machine Learning Algorithms .....	60
4.2.2. Inputs and Outputs of Trained Dataset Parameters .....	60
4.3. Results and Discussion.....	63
4.4. Summary .....	66
5. MILLING PROCESS MONITORING BASED ON INTELLIGENT REAL-TIME PARAMETER IDENTIFICATION FOR UNMANNED MANUFACTURING .....	68
5.1. Parameter identification using PBML algorithms .....	68
5.2. Results and discussion .....	70
5.2.1. Improving accuracy of the milling force model predictions.....	70
5.2.2. Parameter identification .....	72
5.2.3. Experimental verification .....	74
5.3. Production Applications.....	76

5.3.1.	Monitoring and fault detection .....	76
5.3.2.	Parameter optimization .....	76
5.3.3.	Current data from CNC controller .....	77
5.4.	Summary .....	78
6.	PHYSICS-INFORMED TOOL WEAR PREDICTION IN TURNING PROCESS: A THERMO-MECHANICAL WEAR-INCLUDED FORCE MODEL INTEGRATED WITH MACHINE LEARNING .....	80
6.1.	Methodology .....	80
6.1.1.	Modeling of the primary, secondary and third deformation zones .....	80
6.1.2.	Modeling flank wear effect on cutting forces .....	84
6.1.3.	Modeling of turning process .....	85
6.1.4.	Physics-informed machine learning model.....	87
6.2.	Experimental set-up .....	89
6.3.	Results and discussion .....	91
6.3.1.	Improvement of the turning model accuracy through ML .....	91
6.3.2.	Flank wear effect on the cutting forces.....	98
6.3.3.	Reverse ML model for wear length prediction.....	103
6.4.	Summary .....	105
7.	CONCLUSION .....	107
7.1.	Conclusion .....	107
7.2.	Future Research Directions .....	109
	REFERENCES .....	111

## LIST OF TABLES

Table 1. Comparison of simulated cutting forces and experimental results. ....	38
Table 2. Input parameters .....	40
Table 3. Hyperparameters of ML models and their optimum values. ....	41
Table 4. Performance parameters of the ML models for unseen test dataset. ....	42
Table 5. Performance metrics for unseen dataset of Inconel 625 .....	47
Table 6. Thermo-mechanical properties of work-piece materials .....	48
Table 7. Special tools geometrical specifications .....	49
Table 8. Performance metrics for special milling tools .....	49
Table 9. Comparison of various ML algorithms.....	62
Table 10. Hyperparameters of ML models and their optimum values. ....	69
Table 11. Training cutting parameters for improved milling force prediction with ML algorithm.....	71
Table 12. Parameter identification performance of ML algorithms. ....	73
Table 13. Comparison of cutting forces: Mechanistic model vs. experiment tests. ....	93
Table 14. Hyperparameter optimization results.....	95
Table 15. Performance metrics of the PIML algorithms for increasing-decreasing force model. ....	97
Table 16. Performance metrics of the PIML algorithms for decreasing force model...	97
Table 17. Performance metrics of wear-included force model: with and without ML implementation. ....	102
Table 18. Performance metrics of ML models for wear prediction.....	104

## LIST OF FIGURES

Figure 1. Schematic representation of the hybrid ML model. ....	34
Figure 2. The schematic view of the differential milling cutting force directions [118] 35	35
Figure 3. Cutting test setup .....	36
Figure 4. Cutting tools (a) Flat end-mill with 50° helix angle (12 mm) (b) Flat end-mill with 30° helix angle (20 mm) (c) Tapered ball end-mill (d) Serrated end-mill (e) Tapered serrated end-mill .....	36
Figure 5. (a) Filtered Measured data (b) Milling Simulation .....	37
Figure 6. Predicted and measured cutting forces for AL7075-T6. (Axial depth of cut is 4mm, Radial depth of cut is 4mm, Spindle speed is 2000 rpm, Feed rate is 0.25 (mm/rev*tooth)).....	38
Figure 7. Predicted and measured cutting forces for Ti6Al4V (Axial depth of cut is 6mm, Radial depth of cut is 2mm, Spindle speed is 1000 rpm, Feed rate is 0.125 (mm/rev*tooth)).....	39
Figure 8. Regression curves of LSBoost model for cutting forces in X direction: a) Training data, b) Test data .....	43
Figure 9. Regression curves of LSBoost model for cutting forces in Y direction: a) Training data, b) Test data .....	43
Figure 10. Regression curves of LSBoost model for cutting forces in Z direction: a) Training data, b) Test data .....	44
Figure 11. Evolution of hyperparameter optimization process for LSBoost model: a) $F_x$ , b) $F_y$ and c) $F_z$ . ....	45
Figure 12. Statistical error analyses for simulation and hybrid model .....	46
Figure 13 Machine learning based prediction. of Inconel 625 cutting forces .....	47
Figure 14. Serrated endmills (a)Wavelength (b)Amplitude measurements. ....	49
Figure 15. Comparison of serrated tools cutting forces predicted by ML and	

experimental tests. ....	50
Figure 16. The prediction accuracy for out-of-range input. ....	51
Figure 17. The schematic view of the milling cutting force directions [121] .....	55
Figure 18. Correct vs incorrect fault detection of (a) Tool diameter, (b) Teeth number, (c) Tool helix angle and (d) Tool rake angle .....	63
Figure 19. Measurement setup.....	64
Figure 20. One revolution of measured cutting forces. ....	65
Figure 21. One revolution of measured cutting forces. ....	66
Figure 22. Flowchart of the parameter identification process. ....	68
Figure 23. Comparison of the measured cutting forces with simulations and enhanced simulations using PBML. (tool diameter is 16mm, number of teeth is 4, runout is 10 $\mu\text{m}$ , axial and radial depth of cuts are 4mm, feed rate is 0.2 mm/rev.tooth, and spindle speed is 2000rpm).....	72
Figure 24. The statistical error analysis of the identification of a) axial depth of cut (ADOC) and b) radial depth of cut (RDOC). The percentage error between the milling force predictions and the measured values were determined for over 50 different unseen data.....	74
Figure 25. a) Experimental setup for verification b) 3D view and c) top and side views of the test workpiece. ....	75
Figure 26. Comparison of identified & actual axial depth of cut and feed rate.....	75
Figure 27. Comparison of identified and actual radial depth of cut and feed rate.....	75
Figure 28. Feed rate optimization utilizing PBML-estimated resultant cutting forces. .	77
Figure 29. a) Comparison of cutting forces collected by dynamometer and ML- predicted forces using CNC controller data, b) Regression curve for unseen test data (The tooth passing frequency is equal to 66.66 Hz). ....	78
Figure 30. a) Representation of cutting-edge including rake, hone and flank faces, b) 3D tool model considering nose radius effect. ....	81
Figure 31. Different pressure distribution patterns on hone and flank faces.....	82

Figure 32. Schematic representation of flank wear. ....	85
Figure 33. A sample of measured and estimated cutting forces (feed rate: 0.05 mm/rev; depth of cut: 0.5 mm; cutting speed: 100 m/min; nose radius: 0.2 mm; hone radius: 15 $\mu$ m, wear length= 190 $\mu$ m). ....	87
Figure 34. Schematic diagram of the hybrid physics-informed ML model: a) Force prediction in the presence of tool wear, b) Wear length prediction via reverse ML model. ....	89
Figure 35. Experimental set-up, a) Lathe machine, MORI SEIKI, NL1500 b) Tool wear measurement by Nano-focus device. ....	90
Figure 36. a) Hone radius measurement (10 $\mu$ m), b) Nose radius measurement by Nano-focus $\mu$ -surf explorer.....	91
Figure 37. A sample of measured and predicted cutting forces (feed rate: 0.05 mm/rev; depth of cut: 0.5 mm; cutting speed: 100 m/min; nose radius: 0.2 mm; hone radius: 15 $\mu$ m).....	93
Figure 38. Progression of hyperparameter tuning in LSBoost model: a) Tangential force, b) Feed force and c) Radial force.....	95
Figure 39. Regression curves of LSBoost model for: a) tangential; b) feed and c) radial forces.....	97
Figure 40. Model-estimated vs. experimental cutting forces: a) tangential; b) feed and c) radial forces. (feed rate= 0.05 mm/rev, depth of cut= 0.5 mm, cutting speed= 100 m/min, nose radius= 0.2 mm, hone radius= 15 $\mu$ m).....	100
Figure 41. Model-estimated vs. experimental cutting forces: a) tangential; b) feed and c) radial forces. (feed rate= 0.05 mm/rev, depth of cut= 1.5 mm, cutting speed= 100 m/min, nose radius= 0.8 mm, hone radius= 5 $\mu$ m).....	101
Figure 42. Regression curves of LSBoost algorithm for wear-considered force model: a) tangential; b) feed and c) radial forces (wear length varies between 45 $\mu$ m and 300 $\mu$ m).....	102
Figure 43. Regression curves of employed ML models for wear prediction: a) LSBoost; b) RF and c) SVR. (feed rate= [0.05, 0.1, 0.15, 0.2] mm/rev; depth of cut= [0.5, 1, 1.5,	

2] mm; cutting speed= [100, 125, 150] m/min; nose radius= [0.2, 0.4, 0.8] mm; hone  
radius= [5, 15, 30]  $\mu\text{m}$ ). ..... 104

## 1. INTRODUCTION

In recent years, the manufacturing industry has experienced a paradigm shift driven by the principles of Industry 4.0, emphasizing automation, and the integration of smart technologies. This shift has been particularly transformative for machining operations, where precision, efficiency, and real-time monitoring are paramount. The development of advanced simulation models for milling and turning processes, utilizing machine learning (ML) algorithms, has shown promise in enhancing the accuracy of these simulations. Subsequently, these algorithms have been employed to develop sophisticated monitoring systems, significantly improving fault detection, process optimization, and overall manufacturing efficiency.

The motivation for this technological advancement stems from the growing need to improve production efficiency, reduce costs, and maintain high product quality standards in modern manufacturing. One of the critical challenges in milling and turning operations is the deterioration of cutting tool performance due to wear, breakage, and chipping, which can lead to increased production costs, decreased product quality, and longer production cycles. Traditional tool and process monitoring methods, such as direct observation and indirect sensor-based monitoring, often suffer from limitations such as subjectivity, low sensitivity, and lack of adaptability to varying cutting conditions. Therefore, there is a pressing need for more sophisticated and data-driven approaches to tool and process monitoring.

The integration of machine learning algorithms into the simulation of milling and turning processes represents a significant advancement in this field. These ML models can process large amounts of data and learn complex patterns, making them particularly suitable for applications involving multiple factors. By training these models on extensive experimental and simulation data, it is possible to enhance the accuracy of milling and turning simulations significantly. This improvement in simulation accuracy lays a strong foundation for further advancements in process monitoring and optimization.

Using force models, combined with machine learning algorithms, offers a comprehensive understanding of the interaction between cutting tools and workpieces. By calculating

cutting forces for each angular increment of the tool, the model provides accurate predictions that are crucial for optimizing machining parameters, tool geometries, and cutting strategies. The integration of these models with machine learning algorithms enhances the accuracy of simulations, making them a valuable tool for real-time fault detection and process optimization.

Building on the enhanced simulation accuracy, these machine learning algorithms have been employed to develop advanced monitoring systems for milling and turning processes. A significant development in this area is intelligent process monitoring for unmanned manufacturing. Traditional monitoring systems rely heavily on sensory data such as cutting forces, acoustic emissions, or power consumption. However, these methods often fall short due to their time-consuming nature and inability to generalize across different conditions. To overcome these limitations, a hybrid approach that combines physics-based models with machine learning algorithms was developed. This method leverages extensive simulation data to enhance force models, achieving over 96% accuracy in real-time predictions. The results demonstrate the method's applicability in various unmanned manufacturing scenarios, enabling precise parameter identification and fault detection using CNC controller signals.

Accurate prediction of milling forces is crucial for monitoring systems. These predictions inform decisions about power requirements, geometrical accuracy of machined components, stability, and strength of cutting tools. Traditional methods for predicting these forces have relied heavily on empirical and mechanistic models. Empirical models, although straightforward, often lack accuracy due to their reliance on curve-fitted equations that require extensive experimental data. Mechanistic models improve upon this by dissecting the machining process into smaller components, but they too face limitations due to the assumptions and simplifications necessary for their application.

Recent advancements in artificial intelligence (AI) and machine learning (ML) have introduced new methodologies for predicting milling forces. These data-driven approaches offer the potential for higher accuracy but require substantial training data, which can be costly and time-consuming to gather. Integrating ML with mechanistic models provides a promising alternative, leveraging the strengths of both approaches to enhance prediction accuracy while reducing the need for extensive experimental data.

This study proposes a hybrid model that combines physics-based simulations with machine learning algorithms. By training ML models with both simulated and limited experimental data, this approach aims to achieve high prediction accuracy for milling forces across various materials and tool types. The model's effectiveness is validated through tests on materials such as Al7075-T6, Steel 1050, and Ti6Al4V, and is further demonstrated to generalize well to other materials like Inconel 625. The proposed hybrid model not only improves prediction accuracy but also reduces the dependency on extensive experimental datasets, making it a practical solution for real-world industrial applications.

The advancement of unmanned manufacturing has underscored the critical need for intelligent process monitoring to ensure efficiency and quality in machining operations. This necessity was further highlighted during the COVID-19 pandemic, where workforce availability was severely impacted, revealing vulnerabilities in traditional manufacturing setups. Unmanned manufacturing requires robust real-time fault detection systems to maintain operational integrity and product quality.

In another advancement, a novel approach for monitoring tool-related faults in milling processes was developed using process simulation-based machine learning algorithms, for fault detection. This method eliminates the need for costly, time-consuming laboratory tests by utilizing analytical simulation data to train the machine learning models. The proposed approach has demonstrated a 94% accuracy rate in predicting tool-related faults, supported by actual measurement data. This indicates that process simulation-based machine learning algorithms can significantly impact tool condition monitoring and the efficiency of manufacturing processes.

Existing monitoring methods fall into three main categories: experimental, simulation-based, and machine learning (ML)-based systems. Traditional methods relying on sensory data like cutting forces and acoustic emissions are often time-consuming and application-specific, making them difficult to generalize across different conditions. Simulation-based approaches, while beneficial, often lack the precision required due to inherent modeling simplifications. Conversely, data-driven ML systems require extensive datasets to function effectively, which is impractical due to the infrequency of certain fault occurrences.

This study proposes a novel hybrid approach that combines physics-based models with ML techniques to improve real-time parameter identification in milling processes. By integrating mechanistic models with ML algorithms, the proposed method enhances the accuracy of cutting force predictions, thereby facilitating precise fault detection and process optimization. The hybrid model leverages a constrained set of experimental data to refine simulation results, significantly reducing the need for extensive empirical testing.

In the quest for enhanced production efficiency, improved product quality, and reduced manufacturing costs, accurate tool wear prediction remains pivotal. Tool wear significantly affects tool life, surface quality, dimensional accuracy, and the overall economics of machining operations. Traditional methods for predicting tool wear often rely on either experimental data or machine learning (ML) techniques, both of which necessitate extensive and costly wear tests. This dependency on large datasets limits the applicability of ML models in practical industrial settings, confining their use primarily to academic research.

To bridge the gap between research and industrial application, this study introduces a novel physics-informed machine learning (PIML) model for predicting tool wear in turning processes. By integrating an analytical wear-included force model with advanced ML algorithms such as least-squares boosting, random forest, and support vector machines, the proposed approach aims to enhance prediction accuracy while minimizing the need for extensive experimental data. The foundation of the PIML model is a thermo-mechanical turning model that accounts for the effects of flank wear and edge forces on cutting forces. This model's accuracy is further refined through ML, achieving over 97% accuracy on training datasets and 94% on unseen test datasets.

The hybrid PIML model not only enhances the precision of wear predictions but also streamlines the data collection process, reducing the reliance on numerous wear tests. The results highlight the potential of PIML models to revolutionize tool wear prediction by offering a reliable, efficient, and scalable solution for modern manufacturing processes.

Furthermore, the integration of these advanced monitoring systems aligns with the broader trends of digital transformation and the Industrial Internet of Things (IIoT). As manufacturing processes become increasingly interconnected, the ability to gather and

analyze vast amounts of data in real-time will be essential for maintaining competitive advantage. Machine learning algorithms, combined with robust databases and simulation models, provide the tools necessary to harness this data and drive continuous improvement in manufacturing operations.

In conclusion, the integration of machine learning algorithms into tools and process monitoring represents a significant advancement in the field of manufacturing. The ability to accurately predict and detect tool faults, machining parameters, and enhance overall process efficiency is crucial for meeting the demands of modern production environments.

## **1.1. Motivation**

In the current landscape of the manufacturing industry, the principles of Industry 4.0 have ushered in a new era characterized by automation and the integration of smart technologies. This transformation is especially critical in machining operations, where precision, efficiency, and real-time monitoring are vital for maintaining competitive advantage. However, the industry still faces significant challenges that hinder optimal performance and efficiency.

One of the primary challenges in milling and turning operations is the deterioration of cutting tools due to wear, breakage, and chipping, which can lead to increased production costs, decreased product quality, and prolonged production cycles. Traditional methods of tool and process monitoring, such as direct observation and indirect sensor-based approaches, often fall short due to their subjectivity, low sensitivity, and inability to adapt to varying cutting conditions. These limitations underscore the need for more advanced, data-driven monitoring systems that can provide accurate and real-time insights.

The motivation for this research stems from the critical need to address these challenges by leveraging the power of machine learning (ML) algorithms to enhance simulation accuracy and develop sophisticated monitoring systems. These advancements have the potential to significantly improve fault detection, process optimization, and overall manufacturing efficiency. Moreover, the COVID-19 pandemic has highlighted vulnerabilities in traditional manufacturing setups, emphasizing the importance of

unmanned manufacturing and the need for robust real-time fault detection systems to maintain operational integrity and product quality.

## **1.2. Objectives**

### **1. To Develop Hybrid Simulation Models:**

Integrate physics-based simulations with machine learning algorithms to enhance the accuracy of milling and turning process simulations. Train ML models on both experimental and simulation data to capture complex patterns and improve prediction accuracy.

### **2. To Enhance Tool and Process Monitoring:**

Create advanced monitoring systems that utilize ML algorithms to detect tool wear, breakage, and chipping in real-time. Employ force models combined with ML to understand the interaction between cutting tools and workpieces, enabling precise fault detection and process optimization.

### **3. To Implement Real-Time Fault Detection:**

Develop a hybrid approach that combines physics-based models with ML techniques to improve real-time parameter identification and fault detection in milling processes. Leverage extensive simulation data to enhance force models, achieving high accuracy in real-time predictions for unmanned manufacturing scenarios.

### **4. To Minimize Dependency on Extensive Experimental Data:**

Introduce a novel physics-informed machine learning (PIML) model for predicting tool wear in turning processes, integrating analytical wear-included force models with advanced ML algorithms. Reduce the need for extensive experimental datasets by refining simulation results with a constrained set of experimental data, making the approach practical for real-world industrial applications.

### **5. To Align with Digital Transformation and Industrial Internet of Things IIoT Trends:**

Ensure that the developed monitoring systems are compatible with the broader trends of digital transformation and the Industrial Internet of Things (IIoT). Utilize the capability

of ML algorithms combined with robust databases and simulation models to gather and analyze vast amounts of data in real-time, driving continuous improvement in manufacturing operations.

This research aims to bridge several critical gaps identified in the current literature and industry practices:

- **Subjectivity and Low Sensitivity in Traditional Monitoring:**

Traditional methods suffer from subjective interpretation and low sensitivity, limiting their effectiveness. The proposed ML-based monitoring systems aim to provide objective, high-sensitivity detection of tool and process faults.

- **Adaptability to Varying Cutting Conditions:**

Existing monitoring methods often fail to adapt to different cutting conditions. The integration of ML algorithms allows the proposed systems to learn from diverse data sets, enhancing their adaptability and generalizability across various scenarios.

- **Time-Consuming and Application-Specific Sensory Data:**

Current methods relying on sensory data are often time-consuming and application specific. The hybrid approach proposed in this study combines the strengths of physics-based models and ML, reducing the reliance on extensive sensory data and enabling quicker, more accurate fault detection.

- **Dependency on Extensive Experimental Data:**

Traditional ML models require large datasets for training, which can be impractical in industrial settings. The proposed hybrid and PIML models aim to minimize this dependency by leveraging simulation data, thus reducing the need for costly and time-consuming experimental tests.

By addressing these gaps, this research contributes to the advancement of intelligent process monitoring systems, offering practical, scalable solutions that enhance efficiency, reduce costs, and maintain high product quality standards in modern manufacturing environments.

### 1.3. Thesis Overview

The thesis begins with an introduction that sets the stage by highlighting the transformative impact of Industry 4.0 on the manufacturing industry, focusing on machining operations. It underscores the importance of precision, efficiency, and real-time monitoring in milling and turning processes. The motivation for the study is driven by the need to improve production efficiency, reduce costs, and maintain high product quality standards. Traditional tool and process monitoring methods, such as direct observation and indirect sensor-based approaches, are discussed, emphasizing their limitations such as subjectivity, low sensitivity, and lack of adaptability to varying cutting conditions. These limitations highlight the necessity for advanced, data-driven approaches, leading to the outlined objectives of the thesis: developing hybrid simulation models, enhancing tool and process monitoring, implementing real-time fault detection, minimizing dependency on extensive experimental data, and aligning with digital transformation and IIoT trends.

Following the introduction, the literature review chapter provides a comprehensive examination of existing research on milling process monitoring and tool wear prediction. It explores traditional methods, simulation-based approaches, and machine learning (ML) techniques. The review identifies significant gaps in the literature, such as the subjectivity and low sensitivity of traditional methods, the lack of adaptability to different cutting conditions, and the extensive data requirements of ML models. In terms of tool wear prediction, the literature review discusses the limitations of empirical and mechanistic models, which often require extensive experimental data and make simplifying assumptions that can reduce their accuracy. These gaps underscore the need for more sophisticated monitoring systems that integrate the strengths of different approaches.

The third chapter presents a hybrid approach for improving milling force predictions. This approach integrates physics-based simulations with ML algorithms to achieve remarkable accuracy across diverse unseen materials and tool types, including ball end mills and serrated tools. By training ML models on both simulated and limited experimental data, the hybrid model enhances prediction accuracy while reducing the need for extensive empirical testing.

The fourth chapter focuses on developing a smart tool-related faults monitoring system using process simulation-based ML algorithms. This system leverages analytical simulation data to train ML models, achieving high accuracy in predicting tool-related faults without the need for costly and time-consuming laboratory tests. The approach demonstrates significant improvements in tool condition monitoring and overall manufacturing efficiency.

In the fifth chapter, the thesis delves into intelligent real-time parameter identification for unmanned manufacturing. The proposed monitoring system combines physics-based models with ML techniques to improve real-time parameter identification and fault detection in milling processes. By utilizing extensive simulation data, the system enhances force models, achieving high accuracy in real-time predictions and supporting the advancement of unmanned manufacturing.

The sixth chapter addresses tool wear prediction in turning processes, introducing a physics-informed machine learning (PIML) model. This model integrates a thermo-mechanical wear-included force model with advanced ML algorithms to predict tool wear accurately. The PIML model reduces the dependency on extensive experimental datasets, offering a reliable and scalable solution for modern manufacturing processes.

The thesis concludes with a summary of the key findings and their implications for the manufacturing industry. It highlights the contributions of the research in advancing intelligent process monitoring systems, improving prediction accuracy, and reducing costs. The final chapter also discusses potential future work, suggesting areas for further research and development to continue enhancing machining operations through advanced simulation models and machine learning algorithms.

## 2. LITERATURE REVIEW

Milling is a crucial machining process in manufacturing, involving the removal of material from a workpiece using rotary cutters. Effective tool monitoring in milling is essential for optimizing performance, ensuring product quality, and minimizing downtime. This literature review explores recent advancements and methodologies in the milling process and tool monitoring, providing insights into various techniques and their effectiveness.

Milling involves a variety of operations and setups, allowing for the production of complex geometries and high surface finish. The primary parameters influencing milling are spindle speed, feed rate, depth of cut, and tool geometry. According to C. Brecher et al., these parameters must be precisely controlled to achieve desired machining outcomes and avoid tool wear and failure [1].

Tool wear is a critical factor affecting the efficiency and quality of the milling and turning processes. As tools wear, they become less effective, leading to poor surface finish, dimensional inaccuracies, and increased machining costs. K. Jemielniak emphasized that understanding the mechanisms of tool wear, such as abrasion, adhesion, and diffusion, is vital for developing effective monitoring strategies [2].

Tool monitoring can be broadly categorized into direct and indirect methods. Direct methods involve measuring wear using sensors and imaging technologies, while indirect methods infer tool condition from process parameters and signals. Direct measurement techniques include optical and vision-based systems, which use cameras and image processing algorithms to detect wear on the tool surface. S. Y. Liang et al. highlighted that these systems offer high accuracy but can be expensive and complex to implement [3]. Another direct method is acoustic emission (AE), where sensors detect high-frequency sound waves generated during cutting. According to D. E. Dimla, AE is effective in identifying tool wear and predicting tool life [4].

Indirect measurement techniques include vibration analysis, where vibration sensors measure the oscillations of the tool and workpiece. W. T. Chang and S. J. Hsieh found that vibration patterns can effectively indicate tool wear and breakage [5]. Cutting force

monitoring involves measuring the forces in the cutting process, providing insights into tool condition. M. N. Islam et al. demonstrated that variations in cutting forces correlate with different wear states [6]. Temperature monitoring uses thermocouples and infrared sensors to measure the temperature near the cutting zone. T. Moriwaki and M. Shamoto reported that high temperatures are often associated with increased tool wear [7].

Recent advancements in technology have led to the development of sophisticated monitoring systems that integrate multiple sensors and employ machine learning algorithms. Sensor fusion, which combines data from multiple sensors (vibration, AE, force, temperature), can provide a more comprehensive understanding of tool condition. B. Sick showed that sensor fusion enhances the accuracy of tool wear prediction [8]. Machine learning and artificial intelligence techniques, such as neural networks and support vector machines, are increasingly used for tool condition monitoring. R. Teti et al. demonstrated that machine learning models could predict tool wear with high accuracy using sensor data [9].

Despite significant advancements, challenges remain in the field of tool monitoring. These include the need for real-time monitoring, handling large datasets, and ensuring the robustness of monitoring systems in varying operational conditions. X. Li et al. suggested that future research should focus on developing more adaptive and intelligent monitoring systems that can self-correct and improve over time [10].

Effective process monitoring in milling is essential for optimizing machining processes, reducing costs, and ensuring product quality. Advances in sensor technology and artificial intelligence hold promise for the future of tool monitoring, making it more accurate, reliable, and efficient.

## **2.1. Enhanced Milling Forces Estimation**

To design effective machining systems, it is crucial to obtain accurate quantitative predictions of cutting force components in machining operations. These predictions provide a framework for determining the power requirements, machined component geometrical errors or deviations, stability and chatter vibration characteristics, and

strength requirements of cutting tools, jigs, and fixtures. Historically, empirical methods have been used to determine forces in practical machining operations [11–14]. In these methods, the effects of more apparent process variables, such as feed rate, depth of cut, and cutting speed, are related to the experimentally measured average force components using curve-fitted (or empirical) equations [15,16]. Semi-empirical or mechanistic approaches have been used to forecast the force components for specific cutting conditions in actual milling operations, where the radial cut thickness and accompanying forces might fluctuate cyclically during cutter rotation [17,18]. End milling has been studied using a mechanistic approach to anticipate force variations in both rigid and flexible cutter-workpiece systems and to extend its application to forecast associated machine components or surface geometrical defects[19,20]. In addition to predicting forces, the mechanistic approach can also forecast associated machine component deflections and form errors[12,13,21,22]. The cutting force models for the end milling process that have been discussed in the literature can be broadly categorized into four groups: experimental models [23], mechanics-based analytical force models [24,25], mechanistic force models[26], and machine learning-based (ML) data-driven models [27–30]. Analytical models based on mechanics require prior information on shear angles, mean friction angles, chip flow angles, etc., the accurate estimation of which is challenging. Furthermore, the analytical model restricts the practicality of these approaches because it requires numerous experiments for various combinations of work materials, cutter geometry, and cutting conditions. Mechanistic models meticulously forecast cutting forces by dissecting the machining process into precise increments and rotation angles, dissecting each flute individually, and segmenting the end mill axially. The uncut chip area for each element was determined through geometric calculations and correlated with the components of the cutting force using mechanical constants. These constants combine the geometry, material properties of the workpiece, and material of the cutting tool into an empirical relationship that can be obtained by performing a few tests. For a variety of machining processes, Artificial Intelligence (AI)-based algorithms have also been used to estimate cutting forces and other process variables, including surface roughness, tool life, and wear[31–33], etc. Fuzzy logic [34], genetic algorithms, Neural networks (NN) [35] are a few of the several approaches that can be considered examples of AI-based techniques. Tandon et al.[36] expanded the backpropagation algorithm and

feed-forward neural network for predicting cutting forces during milling operations. To learn the behavior of milling process mechanics, the ML model requires extensive training datasets throughout the training (i.e., learning) stage. According to Briceno et al. [37] and Dave and Raval[38], the integration of the design of experiments-based strategy with ML could significantly reduce the number of tests while improving prediction accuracy. When determining the robustness of an ML model, the accuracy and dependability of the datasets used during the learning stage are crucial. Radhakrishnan and Nandan [28] proposed using filtered data in the training of the ML model and created a regression model to remove anomalous datasets from experimental findings. Xu et al.[39] presented a model for predicting the cutting force for the end-milling processes of complex Cutter Workpiece Engagement (CWEs) based on elementary physical priors. The CWE geometry was represented using grayscale images, providing a universal input for ForceNet. With ForceNet, the directional cutting force elements are approximated by elementary neural networks rather than by deep neural networks, which are typically treated as unexplainable black boxes [39]. Furthermore, Xie et al. [40] created images called images of comprehensive geometric processing information (ICGPI), which include comprehensive geometric and processing information in the machining process. The ICGPI is used as an input to design a deep learning network called the Milling Force Convolutional Neural Network (MF-CNN), and the three-dimensional instantaneous cutting forces are the outputs of their model. Moreover, machine learning and neural networks have been used in experimental studies. Kara et al. [41] aimed to predict the cutting forces generated during the orthogonal turning of AISI 316L stainless steel by using artificial neural network (ANN) and multiple regression algorithms. Vaishnav et al. [27] present a method of predicting instantaneous cutting force variation during end milling using the mechanistic model and the supervised neural network. However, in this method, the cutting force was predicted using only simulation data and machine learning algorithms. The outputs of the analytical simulations may have lower accuracy when compared to real measurement data because of certain assumptions made during the analytical modeling of milling simulations. Furthermore, the exact determination of cutting and edge force coefficients requires cumbersome experiments and calibration for utilizing them in milling processes. In relation to the previous studies, Wang et al. [42] proposed a transfer learning NN model to predict the cutting forces of the milling process. In this study, the finite element simulation data were used to train the NN algorithm.

Although the proposed transfer network shows higher accuracy in comparison to ordinary algorithms, data preparation using finite element methods is time consuming and incurs high computational costs for machining simulations.

Based on the literature mentioned above, it is evident that the majority of AI-based data-driven models in this context significantly depend on experimental datasets, whereas only a few studies utilize simulated datasets because of their perceived unreliability. The accuracy of the models relying on experimental datasets is strongly influenced by the size of the training dataset. This represents one of the most challenging limitations of ML models for industrial applications owing to the high cost and time requirements associated with experimental testing. It is important to consider that calibration of the datasets employed in the learning or training process affects the accuracy of the ML model. The datasets produced by machining tests contain a significant number of outliers arising from process disturbances, including vibrations, tool-workpiece defections, and cutting temperatures, which negatively affect the overall prediction accuracy of the algorithm. The application of ML models is constrained by a large number of datasets and noise in the experimental results, making it impossible to deploy them in practice for milling processes. Owing to these restrictions, most ML models discussed in the literature forecast the average cutting forces rather than instantaneous fluctuations, which do not accurately reflect the actual intermittent nature of the milling process. Analytical models have the advantage of being able to estimate instantaneous cutting forces for particular tools and work materials. However, owing to the simplifications and various assumptions during the analytical modeling of the milling mechanics, these models are not very accurate. In this regard, the combination of analytical and data-driven machine-learning algorithms can compensate for the limitations of both types of algorithms.

In this study, a hybrid mechanistic and AI-based force model is proposed that incorporates analytical models into a data-driven ML approach. Such a hybrid model overcomes the limitations of the experimental dataset preparation faced by data-driven AI-based models, simultaneously improving the accuracy of the analytical models. The hybrid ML model has been trained using analytical simulation (along with the cutting parameters) results as inputs and real measured cutting forces as targets. This approach results in increased reliability and accuracy for practical applications in the machining industry. Therefore, a hybrid force model was developed using mechanistic machining models and AI-based

techniques. It is employed to train the machining dataset under various conditions, including different materials, such as Al7075-T6, Steel 1050, and Ti6Al4V, as well as various tool types. Then, the accuracy and generality of this model were assessed in relation to the prediction of the cutting forces under completely different conditions. In addition, this hybrid ML model was employed to predict the cutting forces of unseen conditions, such as various materials (i.e., Inconel 625 superalloy) and different cutting tool types, including end mills with normal and serrated edges with different shapes (i.e., cylindrical and tapered end mills with flat, ball, and round noses).

## 2.2. Milling Process Parameter Identification

The intensive research on monitoring of machining operations is motivated by the ever-increasing interest in enhancing manufacturing efficiency, quality, [43,44] and process optimization as well as realization of unsupervised manufacturing [45]. The importance of unsupervised manufacturing was realized once more by communities at large during the COVID-19 pandemic which caused reduced or completely halted production in many sectors due to unavailability of the workforce in factories [46]. Consequently, in the realm of unmanned manufacturing, identification of real-time anomalies and deviations enabled by process monitoring has gained more attention than ever, making the need for real-time fault source detection capability particularly critical.

The milling process monitoring systems can be divided into three main categories: experimental, simulation, and data-driven machine learning (ML) based systems [44]. Process monitoring traditionally relies on trained systems using sensory data such as cutting force, acoustic emission signals, or power consumption. However, collecting, analyzing such data, and establishing the acceptable limits is time-consuming, and also case and application dependent which cannot be generalized to other [9,47]. In these systems, deviations from accepted margins are detected indicating a problem which is most often attributed to the tool condition, e.g., excessive wear, however *the actual fault sources cannot be identified*. The primary challenge in these methods lies in the time-consuming process of setting margins for all tools and conditions without considering other potential fault sources. Sensory signal levels are influenced by various factors other

than tool sharpness, including material properties, process and machine conditions adding complexity to precision monitoring [48]. Limited research has been conducted in the second category, i.e., in simulation-based monitoring [49]. The simulation-based monitoring approach proposed by Altintas and Aslan [43], relying on predicted and measured forces, could accurately detect tool failure where the sudden peaks in cutting forces are exclusively associated with the tool condition, which might not always be the actual reason. The data-driven monitoring systems rely on analysis of extensive amounts of data collected under various machining states requiring very long development time during which the system will not be effectively functional [27]. This is because of the fact that the occurrence of some of the faults such as machine, process, fixturing or material related issues are relatively less frequent than when compared to the faults for the cutting tools [9]. Therefore, there is a clear need for a solution which can detect fault sources in machining systems accurately in a time efficient manner, both in the development stage and during the process in real-time. In this regard, employing a hybrid approach that combines physics-based model predictions and experimental data using ML can be effective providing rapid and accurate response suitable for real-time applications based on a very limited number of measurements. The proposed method in this work has the potential to overcome the limitations and problems of previous studies.

In this study, a hybrid physics-based machine learning (PBML) algorithm is employed to identify parameters by utilizing a mechanistic linear edge force model. To enhance the accuracy of predicted cutting forces, the model is further refined through the integration of ML algorithms using a constrained set of measured cutting forces. In this regard, a limited number of tests with the conditions commonly employed in industrial applications have been conducted to improve simulation results [50]. Afterwards, enhanced cutting force simulations were utilized to create a substantial database instead of relying on the measured data. Owing to the very fast generation of large amounts of data with this approach, it was possible to develop a large database encompassing a broad range of cutting parameters with fine precision levels in a short time. Subsequently, this database was employed for precise parameter identification through ML techniques in real-time. Therefore, this approach eliminates the necessity for a vast number of measurements yet provides remarkably high prediction accuracy.

The proposed method is validated experimentally on complex geometries with continuous variations in machining conditions where good agreement between predicted and actual values are demonstrated. One of the major and novel advantages of the proposed approach is its suitability to monitoring systems capable of fault source detection with high accuracy in real-time.

### **2.3. Tool Condition Monitoring (TCM)**

These days, modern manufacturing industry are constantly striving to improve production efficiency, reduce costs, and maintain high product quality standards. Cutting tool condition is a key factor that influences these objectives in milling processes. Tool condition monitoring (TCM) is important because it helps to predict and detect tool faults and failure, resulting in reduced machine downtime, improved product quality, and more efficient tool replacement procedures. By applying machine learning algorithms for tool condition monitoring in milling processes, TCM will be more practical by utilizing data-driven techniques.

In subtractive manufacturing, milling is the process by which materials are removed from workpieces with rotating cutting tools in order to achieve the desired shapes and features. It is known that the performance of cutting tools deteriorates over time as a result of wear, breakage, and chipping, which can result in undesirable consequences such as increased production costs, a decrease in product quality, and a longer production cycle. In modern manufacturing, milling process simulations have become an integral part of the process, as they enable engineers to analyze and optimize various aspects of the machining process without having to conduct physical experiments. These simulations are particularly beneficial when using analytical methods for calculating cutting forces because they provide a comprehensive understanding of how the cutting tool interacts with the workpiece. A variety of factors are considered in these models, including tool geometry, workpiece material properties, cutting speed, feed rate, and depth of cut. Mechanic models, oblique cutting theories, and linear edge force models are common analytical methods. The linear edge force model is an analytical method used to predict cutting forces in milling processes. Using this simplified approach, it is possible to estimate the

force experienced during machining by considering the cutting forces acting along the cutting edge of the milling tool. There are certain cases where the linear edge force model is particularly useful, such as when cutting conditions are stable and the cutting-edge engagement with the workpiece is relatively constant.

The linear edge force model [51] is based on the idea that the cutting forces acting on the tool can be represented as a linear distribution of force components along the cutting edge. These force components typically include the tangential force ( $F_t$ ), radial force ( $F_r$ ), and axial force ( $F_a$ ). The model divides the cutting edge into small segments, and the forces acting on each segment are calculated based on the chip thickness, cutting speed, and other relevant parameters.

The linear edge force model offers several advantages in milling process simulations, such as its simplicity and relatively low computational requirements. This model can provide a reasonable estimation of cutting forces, which can be used for optimizing machining parameters, tool geometries, and cutting strategies. However, it is essential to consider the limitations of the linear edge force model, such as its reliance on the assumption of constant cutting-edge engagement and stable cutting conditions.

Traditionally, TCM has been approached through various methods such as direct observation, indirect monitoring using sensors, and statistical process control. While these methods have proven useful to some extent, they often suffer from limitations such as subjectivity, low sensitivity (small changing in input parameters), and a lack of adaptability to varying cutting conditions. Implementing tool condition monitoring in milling operations can lead to several benefits, such as reduced machine downtime, improved product quality, extended tool life, and optimized tool replacement scheduling. Additionally, TCM can be integrated with Industry 4.0 technologies, such as internet of things(IoT) and cloud computing, to enable remote and centralized monitoring of multiple milling machines in a manufacturing facility. In recent years, the emergence of machine learning algorithms has opened new approaches for TCM, offering a more sophisticated and data-driven approach to the problem.

As machine learning algorithms are capable of processing large amounts of data and learning complex patterns, they are particularly suitable for TCM applications involving multiple factors, such as cutting forces, vibrations, and acoustic emissions. In order to

prevent and detect tool failure more accurately and in real-time, machine learning models are trained on simulation data from milling processes. This allows tool maintenance and replacement to be approached in a more proactive manner.

This paper will investigate various machine learning algorithms, such as Multiple Linear Regression, K-Nearest Neighbor (KNN), and Random Forest, in the context of TCM for milling processes. The performance of these algorithms has been assessed in terms of accuracy, sensitivity, and robustness, and provide recommendations for their practical implementation in industrial settings.

Typically, measuring cutting forces serves as an indirect approach for real-time tool condition monitoring [52]. The amplitude of cutting force is used in milling operations as a means of monitoring flank wear. In some studies, online tool condition monitoring systems based on cutting forces has been developed [53]. During the milling process, tool wear was monitored by measuring the average cutting force, revealing that the variation in cutting force consistently increased throughout the machining process. This observation confirmed that cutting tools progressively lost their sharp edges and became worn [54,55]. Artificial Neural Network (ANN) based tool condition monitoring systems were developed using cutting force signals in milling processes to predict flank wear and surface roughness, indicating that cutting force signals increased alongside tool wear [56]. Tool wear in face milling was estimated using the Normalized Cutting Force (NCF) indicator and the Torque-Force Distance (TFD) indicator. The TFD method was found to be superior to NCF, as it remained unaffected by cutting parameters and their interactions [57]. An analysis of wear progression and changes in cutting force was conducted for coated carbide tools. This study illustrates that flank wear was the primary failure mode and had a significant impact on the tool's life [58]. Tool wear in the milling process was monitored using cutting force as the monitoring signal and the Continuous Hidden Markov Model (CHMM) as a diagnostic technique [59]. By tracking the tangential and radial cutting force coefficients during the end milling process, tool wear was monitored. The behavior of these cutting force coefficients was found to be independent of cutting conditions and correlated with tool wear [60]. Tool Condition Monitoring systems (TCMs) for Glass Fiber Reinforced Plastic (GFRP) composite end milling were developed using cutting force signals and the Adaptive Network-based Fuzzy Inference System (ANFIS). The findings confirmed that ANFIS-based feed force data accurately

predicted tool wear [61].

The tool condition was predicted using various decision-making algorithms based on the extracted features. Decision-making algorithms play a crucial role in the development of Tool Condition Monitoring systems (TCMs). A wide range of techniques has been explored to automate TCMs, such as Probabilistic Neural Network (PNN) [62], Support Vector Regression (SVR) [56], Support Vector Machine (SVM) [63], pattern recognition, Artificial Neural Networks (ANN) [64,65], fuzzy logic [66,67], and genetic algorithms [68]. More recently, researchers have employed the Hidden Markov Model [59], ANFIS [61], and Decision Trees [69] to predict tool conditions.

In this work the data mining freeware named 'WEKA' was used for feature classification. Among all classifiers, the random forest classifiers yield the highest classification accuracy. As a result, this study compared tree classifiers to determine the best among them. Other algorithms, such as the KNN algorithm, Naïve Bayes, and LWL, can also be employed if they offer superior classification accuracy. In this case, tree classifiers, including J48, Logistic Model Tree, and Random Forest algorithms are used to classify different milling cutter conditions based on a 10-fold cross-validation. Classification validation is demonstrated using a confusion matrix, as it effectively categorizes distinct tool conditions.

This study presents a novel approach to tool-related fault detection in milling processes by leveraging machine learning algorithms and simulation data. Utilizing machine learning algorithms typically requires a substantial number of tests with various cutting parameters to achieve satisfactory results, which can consume significant time and financial resources. By training machine learning algorithms with milling process simulation data, the drawbacks associated with conducting extensive cutting tests have been effectively circumvented. To accomplish this, the simulations generate a comprehensive milling process database by performing numerous simulations with varying input parameters. Subsequently, the machine learning algorithms were trained using these databases. Finally, this method can identify potential tool-related faults in milling processes. As demonstrated in the results section of this study, the proposed algorithm exhibits a high degree of accuracy in detecting various tool-related faults.

## 2.4. Tool Wear Monitoring

Increasing demands of process automation for un-manned manufacturing have attracted many researchers to the field of online monitoring of machining processes. Extensive research is underway globally in the area of online tool condition monitoring systems (TCMS) [48,70], highlighting its critical role in modern industrial setups. Among the challenges faced, tool wear emerges as the most undesirable characteristic of machining processes. It adversely affects tool life, which is paramount in metal cutting because of its direct impact on the surface quality of the machined products, their dimensional accuracy, and consequently, the economics of machining operations. This underscores the necessity of tool wear monitoring not just as a routine procedure, but as an essential strategy to maintain productivity, tool life, and product quality. By effectively monitoring tool wear, manufacturers can prevent extensive damage to machine tools, reduce costly downtime, and minimize the number of scrapped components. Therefore, developing and implementing precise methods for cutting tool wear sensing are crucial, ensuring the optimum use of cutting tools and maintaining operational efficiency and reliability.

Tool wear measurements can be categorized into two primary types: direct and indirect measurement methods. Direct measurement techniques are one of the primary methods used to assess tool wear. These include using toolmaker's microscopes [71], which magnify worn areas for accurate measurement, and optical microscopy, which allows for close observation of the wear on the cutting edges. Additionally, scanning electron microscopy (SEM) can provide detailed images of tool surfaces at a microscopic level, enabling precise wear analysis [72]. However, indirect measurement techniques are also popular due to their non-invasive nature. Changes in cutting forces, for example, can be an indicator of tool wear. As the tool wears, the forces required to cut the material increase [73–75]. Monitoring the power consumption of a machine tool provides indirect indications of tool wear; more power is generally needed as the tool degrades [76]. Vibration analysis [77,78] based on short-time Fourier transform (STFT) and acoustic emission sensors are other popular methods where increased vibration or changes in the high-frequency noises produced during the cutting process can suggest tool degradation

[79,80]. Acoustic emission (AE) sensors operate by detecting the high-frequency energy waves generated when materials undergo deformation or fracture. These sensors capture the sound waves emitted during processes like cutting, allowing for the monitoring of tool conditions and the early detection of wear or failure [81]. Choosing the most preferred sensor for indirect tool wear measurement largely depends on the specific application, machining conditions, and the type of information required. Among the presented sensors, acoustic emission and force sensors are utilized more commonly. Unlike acoustic emission sensors, which can be sensitive to background noise and may require complex setups to differentiate relevant signals, force sensors are generally more robust and less affected by external noise. This makes them particularly effective in industrial environments where numerous disturbances might otherwise complicate data accuracy. In many industrial applications, the detection of significant wear before tool failure is more critical than the early detection of minor wear. Force sensors [82,83] provide this by measuring changes in cutting forces that directly correlate with the wear state of the tool, offering a practical approach to preventing catastrophic failures and optimizing maintenance. In this study, the impact of tool wear on cutting forces was explored, and force sensors were utilized for experimental validation.

Tool wear modeling and prediction involve various approaches, each designed to anticipate the condition and tool life under different operating circumstances. The main tool wear modeling approaches which are utilized in industries are empirical, finite element analysis (FEA)-based, analytical, statistical and machine learning (ML) models. Empirical models are based on experimental data and observations [84,85]. These models often use simple mathematical formulas derived from historical tool wear data to predict future wear. They are straightforward but can be limited by their dependence on specific measured conditions. FEA-based models emphasize mechanisms of wear, such as abrasion, adhesion, and diffusion, to model the tool wear process accurately. These models include the interactions between the tool and workpiece at a microscopic level, such as using FEA to predict stress and strain patterns that influence wear [86,87]. Moreover, Yen et al. [88] utilizes FEA to calculate the increase in flank wear width after a specified cutting time, providing a detailed analysis of wear rates and progression based on simulated cutting conditions. However, this method leads to high demands on computational resources and extended processing times, which can be impractical for

real-time applications. The accuracy of FEA predictions heavily depends on the material models used. For tool wear, the material behavior under high stress, high temperature, and high strain rates must be accurately modeled, which can be challenging due to the lack of reliable material data under such extreme conditions. Huang et al. [89] developed a comprehensive model for predicting cutting forces in hard turning by extending a two-dimensional mechanistic force model to consider three-dimensional chip formation considering wear effects. The model incorporates the impact of flank wear by adapting a worn tool force modeling approach to three-dimensional contexts, considering process parameters like low feed rates, small depths of cut, and large tool nose radii. Their model, however, did not account for the third deformation zone and the influence of edge forces. Additionally, it only considered specific machining conditions, such as hard turning. Some other studies have been done based on analytical tool wear estimations [90]. Analytical cutting tool wear prediction based on the multi-stage Wiener process was presented by Wang et al. [91]. In this study, the degradation rates of cutting tools are considered to change in three stages based on the typical cutting tool wear curve model. The parameter estimation for each stage of the cutting tools' degradation processes is independently carried out using the Expectation-Maximization (EM) algorithm. Another method for tool wear modeling is prediction using data-driven ML algorithms [92,93]. Gouarir et al. [32] presented a ML system which is used to monitor the progression of tool flank wear utilizing force sensors. In this study, the experiments were conducted using dry machining conditions with an uncoated ball endmill and a stainless-steel workpiece. Predictions from the ML model were derived from a comprehensive database that includes a huge number of experiments. Moreover, Li et al. [94] introduced a tool wear prediction model that utilized an Improved Particle Swarm Optimization (IPSO) technique combined with a Convolutional Neural Network (CNN) and a Bidirectional Long Short-Term Memory (BiLSTM) network. One of the most significant disadvantages of predicting tool wear using ML algorithms is the heavy reliance on large volumes of high-quality data for training which are time consuming. Machine learning models, particularly those that are complex, require diverse and extensive datasets to develop accurate predictions, and any deficiencies in data quantity or quality can lead to poor model performance and unreliable predictions. Collecting these data involves extensive measurements that are both costly and time-consuming.

In this study, a hybrid physics-informed machine learning model (PIML) was developed to address the identified challenges. This innovative approach not only enhances the precision of the analytical model but also eliminates the extensive data collection typically required by conventional data-driven ML algorithms. To achieve this, initially a thermo-mechanical force model was established to calculate the cutting forces in the turning process, considering the effects of tool wear, nose radius, and hone radius. The advantages of the proposed model compared to previous ones include accounting for local pressure and shear stress distribution along the cutting edge in various deformation zones. This provides a more nuanced understanding of how the material behaves during cutting, potentially leading to more accurate predictions of cutting forces. Also, the model accounts for the effect of edge forces and flank wear on the cutting forces. This is a significant improvement, as edge forces and flank wear can significantly impact the cutting process. By first enhancing the model's accuracy with PIML algorithm, it was employed to create a training dataset for a new complementary reverse ML model, to predict tool wear based on the cutting forces, machining parameters and tool geometry. This streamlined tool wear prediction by eliminating the need for extensive and resource-consuming experimental testing. The final phase of this study involved the experimental validation of the method, which confirmed a robust agreement between the modeled results and the actual experimental data, thus underscoring the effectiveness of this hybrid approach in enhancing predictive reliability and reducing the need for extensive data collection typically required by traditional ML models in tool wear analysis.

### **3. IMPROVING MILLING FORCE PREDICTIONS: A HYBRID APPROACH INTEGRATING PHYSICS-BASED SIMULATION AND MACHINE LEARNING FOR REMARKABLE ACCURACY ACROSS DIVERSE UNSEEN MATERIALS AND TOOL TYPES**

#### **3.1. Machine Learning Algorithms (ML)**

##### **3.1.1. Support Vector Regression (SVR)**

Support Vector Machines (SVM) in their contemporary manifestation can be traced back to the collaborative efforts of Vapnik and colleagues at AT&T Bell Laboratories during the 1990s. Initially introduced for binary classification challenges [95], SVM swiftly garnered recognition within the machine learning community, notably distinguished by its accomplishments in Optical Character Recognition (OCR). Support Vector Machines are fundamentally based on the identification of a hyperplane that serves as the decision boundary. These hyperplanes are determined by the resolution of the quadratic programming problem. The optimization problem aims to find the hyperplane that maximizes the margin while guaranteeing the accurate classification of the maximum number of training points. Notably, the so-called kernel trick, a technique recognized for mitigating computational costs, facilitates the application of SVM, even in scenarios characterized by nonlinearity. The efficacy of SVM extends beyond binary classification problems, demonstrating noteworthy performance in diverse machine learning applications. In instances of nonlinear classification challenges, the method's proficiency remains robust, underscoring its versatility.

Support Vector Regression (SVR), which shares a foundational theoretical framework with SVM, is a supervised learning paradigm embraced particularly in scenarios where the target variable assumes numerical values [47,96]. The objective of SVR is to find a function that approximates the relationship between the input features (independent variables) and target output (i.e., the dependent variable) while minimizing the prediction error. This function approximation problem is also considered as a quadratic optimization problem, which aims to find a hyperplane that, maximizes the margin while still ensuring

that the deviations of the data points from the hyperplane (within a certain range, determined by the  $\varepsilon$  parameter) are minimized. In the context of SVR, the margin includes a tolerance margin (managed by the  $\varepsilon$  parameter) that allows points to fall inside the margin without incurring penalties in addition to the distance between the hyperplane and the support vectors. Because of this flexibility, SVR may concentrate on fitting most of the data while allowing some points to stray within a predetermined range. The solution of the following optimization model yields the approximated function used in the regression:

$$\begin{aligned}
 & \min_{w,b,\xi,\xi^*} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) \\
 & \text{s. t.} \\
 & y_i - (w \cdot \phi(x_i) + b) \leq \varepsilon + \xi_i \\
 & (w \cdot \phi(x_i) + b) - y_i \leq \varepsilon + \xi_i^* \\
 & \xi_i, \xi_i^* \geq 0
 \end{aligned} \tag{1}$$

where  $w$  is the weight vector which is perpendicular to the separating hyperplane (i.e., the decision boundary),  $b$  is the bias term,  $\xi_i$  and  $\xi_i^*$  are the slack variables,  $\phi(x_i)$  is the feature vector transformed into a higher dimensional space and " $\cdot$ " is the dot product. The solution of this optimization problem can be obtained analytically only for small sample sizes and for most of the real-world cases various numerical techniques are used to determine the optimal  $w$  and  $b$ , and for an unknown data vector  $u$  (i.e., the test data), the dependent target output is calculated as  $y = w \cdot u + b$ . For further details regarding SVR, please refer to [97].

### 3.1.2. Least Square Gradient Boosting Algorithm (LSBoost)

In this study, we incorporate two distinct ensemble learning methodologies into our analytical framework. Specifically, we employed the Least Square Gradient Boosting (LSBoost) and Random Forest (RF) algorithms. Ensemble learning methods, distinguished by the integration of multiple foundational models known as base learners, are prevalent in contemporary machine-learning paradigms. This paradigm draws inspiration from the theoretical underpinnings articulated in Condorcet's Jury Theorem

dating back to 1785 (for a detailed discussion on the history and assumptions of the theorem, please refer to [98]). The fundamental premise lies in the recognition that while an individual weak learner is susceptible to misguidance, the collective robustness of an ensemble, comprising such weak learners, serves to mitigate susceptibility to erroneous outcomes. In other words, a crowd is smarter than the individuals that are in the crowd.

Various ensemble learning algorithms, such as randomization, bootstrap aggregation (i.e., bagging), boosting, and stacking, have been proposed to enhance the predictive performance and resilience in machine learning applications by leveraging the combined insights of diverse models. Specifically, boosting strategically addresses the identification and rectification of errors made by the existing models through an iterative generation process. Each new model is influenced by the mistakes of its predecessors, assigning additional weights to previously misclassified observations. The algorithm introduces a nuanced dimension to model construction, allowing subsequent models to specialize as experts in addressing misclassified cases. This diversification is achieved through a voting or averaging mechanism, in which the ensemble's collective wisdom is harnessed, with the influence of each model being differentially weighted. This approach often yields a transformative impact on algorithmic performance, emphasizing collaborative learning from diverse models to enhance the predictive accuracy and overall robustness.

The concept of boosting weak learners to create a strong learner originated in Kearns and Valiant [99], and the first polynomial-time boosting algorithm was introduced by Schapire [100]. In the original boosting algorithm, the dataset is partitioned into three sets ( $D_1, D_2, D_3$ ). The first classifier ( $h_1$ ) is trained using the first dataset ( $D_1$ ). Subsequently, the second dataset ( $D_2$ ) serves as the test data for the first classifier ( $h_1$ ). Misclassified instances from the training data ( $D_1$ ) and correctly classified instances from the test data ( $D_2$ ) are then used to train the second classifier ( $h_2$ ). The third dataset ( $D_3$ ) is employed as the test data for both the first and second classifiers ( $h_1$  and  $h_2$ ). Instances where these classifiers disagree are identified, and the third classifier ( $h_3$ ) is trained exclusively on these cases. For the new test data, the first two classifiers are initially utilized, and if they agree, their decision is accepted. If they disagree, the final decision is determined using the third classifier.

Subsequently, Freund and Schapire [101] introduced one of the most famous boosting algorithms, AdaBoost, an abbreviation for Adaptive Boosting that refines the boosting

process by repeatedly using the same training set. AdaBoost extends the methodology to combine an arbitrary number of base learners, eliminating the original three-set constraint and showcasing the adaptability and scalability of the boosting framework. In brief, AdaBoost assigns weights (which correspond to the probability of consideration by the current weak learner) to the training instances and adjusts these weights during each iteration to focus on examples that are misclassified by the current set of weak classifiers. The final model is a weighted sum of the weak classifiers, where the associated weights are determined based on the predictive performance of the classifiers.

Friedman [102] extended the AdaBoost framework by introducing Gradient Boosting Machines, which are commonly referred to as gradient boosting. Given a dataset  $\{(x_i, y_i)\}_{i=1}^N$ , conceptualizes the learning problem as an optimization model, where the primary objective is to minimize a differentiable loss function, denoted as  $L(y_i, F(x))$ . In regression scenarios, the loss function is typically represented by  $0.5 * \text{Mean Squared Error (MSE)}$ , which is defined as half of the sum of squared residuals ( $r$ ), expressed as the difference between the observed value ( $y_i$ ) and the predicted value ( $\hat{y}_i = F(x)$ ). Note that, in the loss function MSE is scaled with 0.5 due to mathematical convenience. LSBoosting stands out as a prominent gradient boosting algorithm tailored for regression tasks when the loss function is  $0.5 * \text{MSE}$ .

Gradient Boost, with  $M$  stages, incorporates a new weak learner ( $h_m(x; \theta)$ ) at each stage to improve the loss function through a gradient descent-like procedure. Notably, in Gradient Boost, the target output for the weak learner  $h_m(x)$  is not the original output of the training data ( $y_i$ ) but rather the residuals ( $r_i$ ). Consequently, for each training data,  $x$ , the target output (i.e., the predicted value) at stage  $m$  is determined based on this residual information as follows

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x; a_m) \quad (2)$$

where  $h_m(x; a_m)$  corresponds to the output of the  $m^{\text{th}}$  weak learner trained for the pseudo residuals ( $r_m$ ) obtained for the  $m^{\text{th}}$  stage,  $a_m$  correspond to the parameters of the learner) and  $\gamma_m$  is the scaling weight (referred to as the learning rate or step size) associated with  $h_m(x; a_m)$ . It is important to note that if  $0.5 * \text{MSE}$  is employed as the loss function, then  $r_m$  are indeed the residuals. However, for other loss functions, like MSE,  $r_m$  would be 2

times the residuals, hence the term pseudo residuals. In contrast to the AdaBoost algorithm, where weights are determined by focusing on misclassified instances, the Gradient Boost algorithm updates weights by calculating the negative gradient of the loss function with respect to the prediction for each instance. This characteristic gives the algorithm its name—Gradient Boosting. It's worth mentioning that when the loss function is  $0.5 \cdot \text{MSE}$ , negative gradient becomes  $r_{im}$ .

Algorithm 1 presents the steps of the LSBoost. It initiates by assigning  $F_0(x)$ , that minimizes the loss function, as the prediction for each instance (line 1). Since the loss function for LSBoost involves minimizing MSE (or half of it), the prediction  $F_0(x)$  is set to be the mean of the output values of the training data ( $F_0(x) = \bar{y}$ ). Subsequently, a loop is initiated to execute the following steps for each stage  $m$  (line 2). The loop begins with computing the negative gradients ( $r_{im}$ ) for each data point ( $x_i$ ). Note that since the loss function is half of MSE, the gradient with respect to the predicted values is the negative of the residual ( $r_i = y_i - \hat{y}_i$ ) (line 3). Next, a weak learner is trained to fit the determined negative gradients ( $h_m(x; a_m)$ ) (line 4). The optimal step size  $\gamma_m$  is determined which minimizes the lost function (line 5) and ultimately the target outputs are updated (line 6).

*Algorithm : Least Square Gradient Boosting*

$$1: \quad F_0(x) = \underset{F(x)}{\operatorname{argmin}} \sum_{i=1}^N L(y_i, F(x_i)) = \bar{y}$$

2: *For*  $m = 1$  to  $M$  *do*

$$3: \quad r_{im} = - \left[ \frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \right]_{F(x)=F_{m-1}(x)} = y_i - F_{m-1}(x_i) \quad \forall i = 1, \dots, N$$

$$4: \quad a_m = \underset{a}{\operatorname{argmin}} \sum_{i=1}^N [r_{im} - h_m(x_i; a_m)]^2$$

$$5: \quad \gamma_m = \underset{\gamma}{\operatorname{argmin}} \sum_{i=1}^N [y_i - (F_{m-1}(x) + \gamma h_m(x_i; a_m))]^2$$

$$6: \quad F_m(x) = F_{m-1}(x) + \gamma_m h_m(x; a_m)$$

- 7: End For
- 8: End Algorithm

The boosting framework was then expanded by scholars and industry professionals, giving rise to well-known and powerful algorithms like XGBoost [103] , LightGBM [104] and others that are used in many different applications.

### 3.1.3. Random Forest (RF)

The Random Forest algorithm employs a combination of diverse randomization techniques to create a resilient ensemble of decision trees, enhancing the overall robustness of the model. Decision trees, fundamental in supervised learning for tasks such as classification and regression, have a rich history, with the earliest regression trees introduced as Automatic Interaction Detection (AID) by Morgan and Sonquist [105]. Notable among tree-based learning algorithms are Classification and Regression Trees (CART) by Breiman et al. [106] and C4.5 by [107], which is an extension of the Iterative Dichotomizer version 3 (ID3) [108]. While decision trees are esteemed for their interpretability, speed, and widespread software availability, their predictive performance is often considered suboptimal.

The evolution of decision trees has been guided by a quest for improved performance and robustness. As decision trees can exhibit variability with different training samples, the manipulation of data introduces diversity in generating decision trees, which lays the groundwork for ensemble learning. Breiman , which introduced Bootstrap Aggregating, leverages this characteristic to create an ensemble by training each tree on a randomized subset of the data. Concurrently, Ho [110] advocated for a random selection of features when growing trees, enhancing diversity. Amit and Geman [111] extended the evolution by using a random subset of features *at each node*, while Dietterich [112] proposed selecting *one of the best* features randomly among a set of promising features at each node. Breiman [109] suggested injection of random noise and creating new features during tree growth.

All these attempts of amalgamating multiple trees with varied perspectives opted for

developing a more potent and accurate predictive model. The primary incentive behind this lies in the recognition that, while decision trees may be considered weak learners individually, their combined influence within the ensemble synergistically enhances the overall predictive capability and robustness of the model. Breiman [113] introduced the Random Forest (RF) algorithm which encapsulates the evolution of decision trees into a robust ensemble model. RF algorithm builds  $M$  decision trees, each grown from a subset ( $D_m$ ) sampled with replacement from the training data. At each node, the algorithm selects the best split from a random subset of  $F$  features. Each tree grows to its maximum extent without pruning, contributing to the diversity of the ensemble. The final prediction is made by aggregating the outputs of all  $M$  trees, employing a majority vote for classification or averaging for regression. Below is the RF algorithm:

*Algorithm: Random Forest*

*Given a training data  $D$*

1. For  $m = 1$  to  $M$  do:
  - 1) Build subset  $D_m$  by sampling with replacement from  $D$
  - 2) Learn tree  $T_m$  from  $D_m$ 
    - 1) At each node:
      - Choose best split from random subset of  $F$  feature
    - 2) Each tree grows to the largest extend, and no pruning
2. Make predictions according to majority vote (for classification) or mean (for regression) of the set of  $m$  trees.

This algorithm introduces randomness in two ways: by randomly sampling features at each split and by training multiple trees on different bootstrap samples of the training data. In conclusion, Random Forest stands as a testament to the evolution from traditional decision trees [114], incorporating randomness and diversity to create a powerful ensemble learning method that excels in handling complex datasets and achieving high predictive performance [107].

#### **3.1.4. Hyperparameters Optimization**

A machine learning model's performance is strongly affected by hyperparameter values. Among different optimization algorithms, Bayesian optimization, grid search and random

search are the most widely used techniques for hyperparameter optimization [115,116]. In grid search, an optimal set of hyperparameters is chosen by taking into account all possible combinations of hyperparameter values. In spite of covering the entire search space, this approach is computationally expensive and inefficient since the number of models to be trained increases exponentially with the number of hyperparameters. Grid search is therefore suitable for ML models with one or two hyperparameters to optimize. In the random search, instead of testing each hyperparameter individually, random combinations of the hyperparameters are examined, resulting in a faster response time. However, the obtained solution does not necessarily represent the best possible one. Both the random search and the grid search algorithms are blind methods, meaning that the previous trial data is not used to select the next hyperparameter values. As opposed to these techniques, Bayesian optimization keeps track of all prior trials to select the next set of hyperparameters, instead of blindly searching the domain space. Therefore, in the present study the Bayesian optimization algorithm was implemented for hyperparameter optimization of the ML models. This algorithm includes two key elements: a probabilistic model for the objective function, and an acquisition function to assess the quality of an evaluation point and determine the subsequent query point. To optimize the objective function, an initial probability distribution (the prior) is generated as the starting point of the optimization process. Whenever an observation of the objective function  $f(x)$  is made at an evaluation point  $x$ , the distribution of probable functions (now known as posterior) narrows. In a subsequent step, using the posterior distribution, an acquisition function is constructed to identify the upcoming inquiry point. As a result, a probabilistic model produces a Bayesian posterior probability distribution which defines the potential  $f(x)$  values at  $x$ . Following the observation of  $f(x)$  at a new candidate point, a new posterior distribution is generated. The acquisition function then selects the next query point based on the current posterior distribution. Among various probabilistic models for objective function estimation, Gaussian Process Regression (GPR), a kind of kernel-based model, has been extensively used. The prior distribution of objective function is a Gaussian process with covariance kernel function  $k(x_i, x_j | \theta)$  and mean  $m(x | \theta)$ .

Under the assumption of additional Gaussian noise with variance  $\sigma^2$  in the observations, the prior distribution will have a covariance of  $K((X, X | \theta) + \sigma^2 I)$ . By determining the kernel parameter  $\theta$  and noise variance  $\sigma^2$ , a GPR model can be fitted to the observations.

In this study ARD Matérn 5/2 kernel function was used for Bayesian optimization algorithm, as explained by equation 3 [117]:

$$k(x_i, x_j | \theta) = \sigma_f^2 \exp(-\sqrt{5}r) \left( 1 + \sqrt{5}r + \frac{5}{3}r^2 \right) \quad (3)$$

$$r = \sqrt{\sum_{m=1}^d \frac{(x_{im} - x_{jm})^2}{\sigma_m^2}}$$

The second essential component of the Bayesian optimization algorithm is the acquisition function, which is used to judge the “merit” of a point  $x$  based on its posterior distribution. In this study, the 'expected-improvement' function was used as the acquisition function, as explained by equation 4:

$$EI(x, Q) = E_Q(\max(0, m_Q(x_{best}) - f(x)) \quad (4)$$

### 3.2. Hybrid Milling Force Prediction Methodology

As discussed in previous sections, the main aim of this study is increasing the accuracy of the mechanistic milling models. For this purpose, machine learning algorithms were trained with dataset considering the cutting parameters and corresponding simulation data as inputs, and real measurement as target values. Afterwards, the improved cutting force predictions are obtained using trained machine learning model. In this regard, different ML algorithms (such as random forest, LSBoost and support vector regression) were implemented, and the prediction accuracy of these models were evaluated in terms of the RMSE and  $R^2$  values. It should be mentioned that before regression analysis, the hyperparameters optimization was performed using Bayesian optimization method. Furthermore, the developed models were used to enhance the simulation accuracy of a completely different material. For this purpose, the ML models were trained by dataset of Aluminum 7075-T6, Steel 1050 and Ti6Al4V. Meanwhile, the trained models were employed to predict the cutting forces of Inconel 625, as an unseen dataset. In addition, this method was utilized for special milling tools including cylindrical, and tapered end mills with flat, ball and round noses. A schematic representation of the proposed ML model is illustrated in Figure 1.

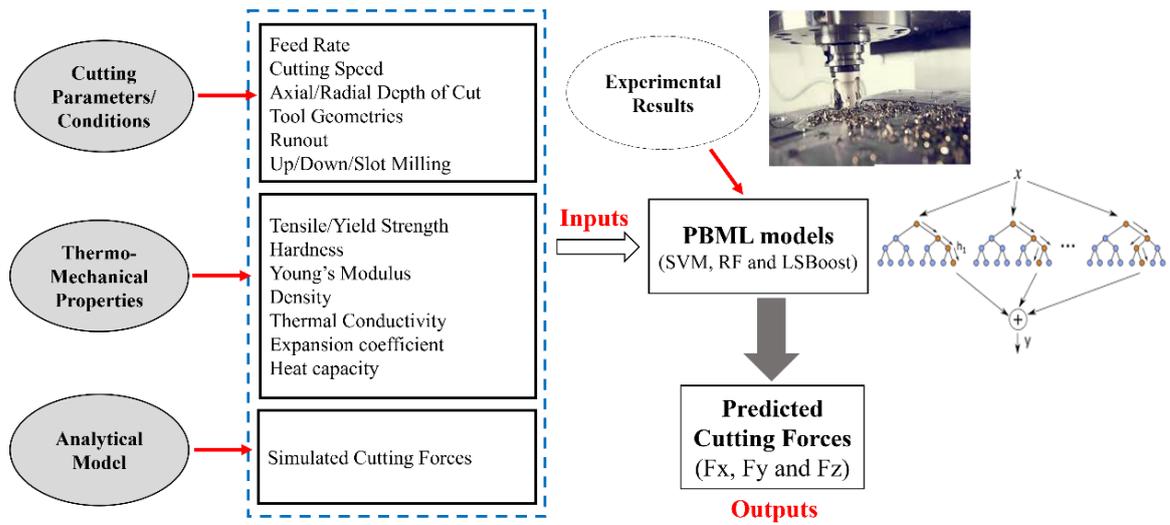


Figure 1. Schematic representation of the hybrid ML model.

The simulated cutting forces used for ML training were obtained utilizing linear edge force model, as described in the following section.

### 3.2.1. Milling Force Modeling

As previously discussed in the preceding section, the first step in simulating cutting forces during milling with various types of end mills involves modeling the mechanics of milling. This study utilizes a general geometric model for commonly used end mills proposed by Tehranizadeh et al. [118]. The models considered in this research encompass cylindrical and tapered end mills with flat, ball, and round nose shapes. Additionally, various types of cutting edges, both straight and serrated, with circular, sinusoidal, and trapezoidal serration waveforms, are employed in this investigation. To formulate milling forces in this study, the linear edge force model [51] is employed. The initial step in the calculation procedure involves acquiring differential forces in the axial, radial, and tangential directions for every tooth on each axial element at any angular position of a cutting tool. The chip thickness in each element of each tooth is obtained using the method described in [118].

The cutting force coefficients,  $K_{rc}$ ,  $K_{tc}$  and  $K_{ac}$ , utilized in the calculation of cutting forces, are determined by applying orthogonal cutting data and employing the oblique

cutting transformation method described by Budak et. Al (1996). This method takes into account the obtained local chip thickness and local cutting angles using the approach detailed by [118,119].

The total forces in  $x, y, z$  directions for angular orientation of the tool can be obtained by summation of the elemental differential forces:

$$\begin{aligned}
 F_x(\varphi) &= \sum_{z=0}^N \sum_{j=1}^{N_t} [-dF_r(i, j, \varphi_{ij}) \sin(\varphi_{ij}) \sin(\kappa_{ij}) \\
 &\quad - dF_t(i, j, \varphi_{ij}) \cos(\varphi_{ij}) \\
 &\quad - dF_a(i, j, \varphi_{ij}) \cos(\kappa_{ij}) \sin(\varphi_{ij})] \\
 F_y(\varphi) &= \sum_{z=0}^N \sum_{j=1}^{N_t} [-dF_r(i, j, \varphi_{ij}) \cos(\varphi_{ij}) \sin(\kappa_{ij}) \\
 &\quad + dF_t(i, j, \varphi_{ij}) \sin(\varphi_{ij}) \\
 &\quad - dF_a(i, j, \varphi_{ij}) \cos(\kappa_{ij}) \cos(\varphi_{ij})] \\
 F_z(\varphi) &= \sum_{z=0}^N \sum_{j=1}^{N_t} [dF_r(i, j, \varphi_{ij}) \cos(\kappa_{ij}) - dF_a(i, j, \varphi_{ij}) \sin(\kappa_{ij})]
 \end{aligned} \tag{5}$$

where  $dF_r(i, j, \varphi_{ij})$ ,  $dF_t(i, j, \varphi_{ij})$  and  $dF_a(i, j, \varphi_{ij})$  are the differential forces in radial and tangential and axial directions, for each tooth ( $j$ ) on each axial element ( $i$ ) at any angular position ( $\varphi_{ij}$ ) of a cutting tool (Figure 2).  $\kappa_{ij}$  is the axial immersion angle,  $N_t$  and  $N$  are the number of teeth and axial elements respectively.

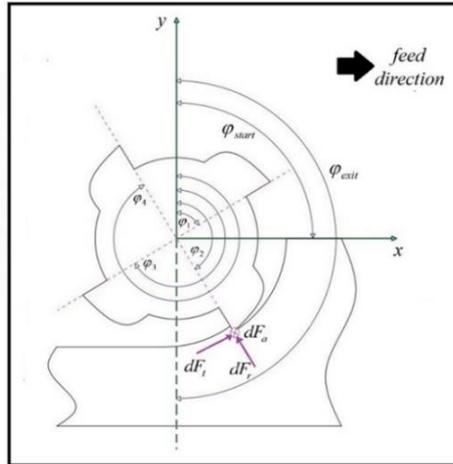


Figure 2. The schematic view of the differential milling cutting force directions [118]

Furthermore, experiments are essential for obtaining cutting forces, which can be

compared to simulation results to enhance their accuracy. Figure 3 depicts the milling process test setup, where cutting forces were measured using a Piezo-electric Dynamometer 9257BA, an amplifier, and the NI USB-6259 data acquisition system. These tests encompass various materials (Aluminum 7075-T6, Steel 1050, Ti6Al4V, and Inconel 625) as workpieces, along with different types of end mills. Figure 4 shows different cutting tools which have been utilize for conducting measurements.



Figure 3. Cutting test setup

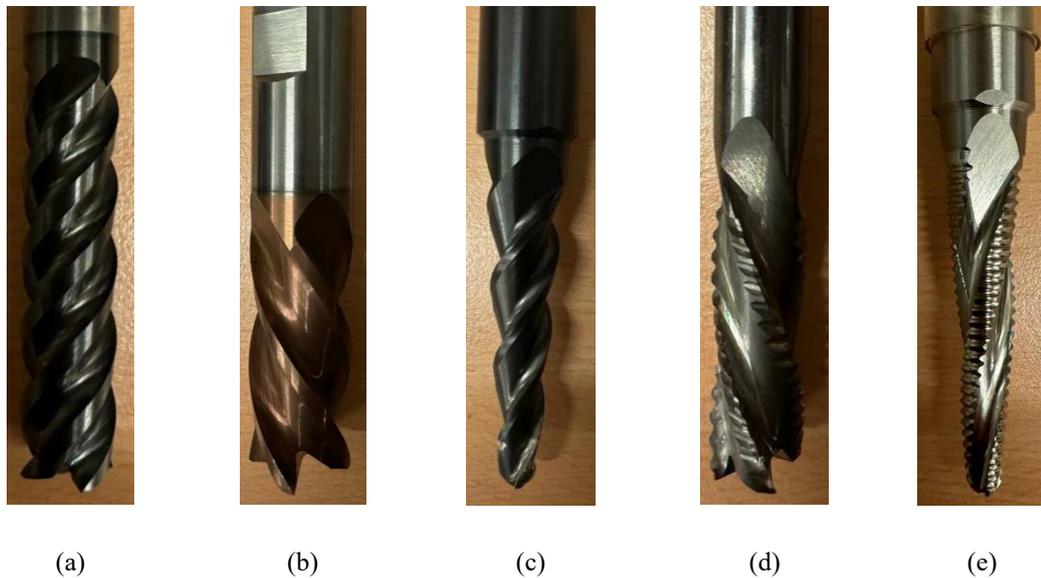


Figure 4. Cutting tools (a) Flat end-mill with 50° helix angle (12 mm) (b) Flat end-mill with 30° helix angle (20 mm) (c) Tapered ball end-mill (d) Serrated end-mill (e) Tapered serrated end-mill

### 3.2.2. Data Preparation

The accuracy of any machine learning model is greatly influenced by the quality of the input data that forms the basis of the model. Data preparation is the first step in the machine learning process. The raw data obtained from the dynamometer must be filtered for this purpose. The air-cutting portions of the data need to be removed in the first stage. In the next step, the cutting force relating to the specific angular position of both simulation and measured data should be synchronized.

For this, the starting point for each simulation and measured dataset is the peak force in one revolution in all directions (i.e., X, Y and Z), as illustrated in Figure 5. Machine learning algorithms were trained using only one revolution of measured data and simulation, for each cutting condition of the milling process. Finally, the dataset was scaled to have zero mean and unit variance. The purpose of feature scaling is to ensure that all features are on a comparable scale, making it easier for most ML algorithms to process them.

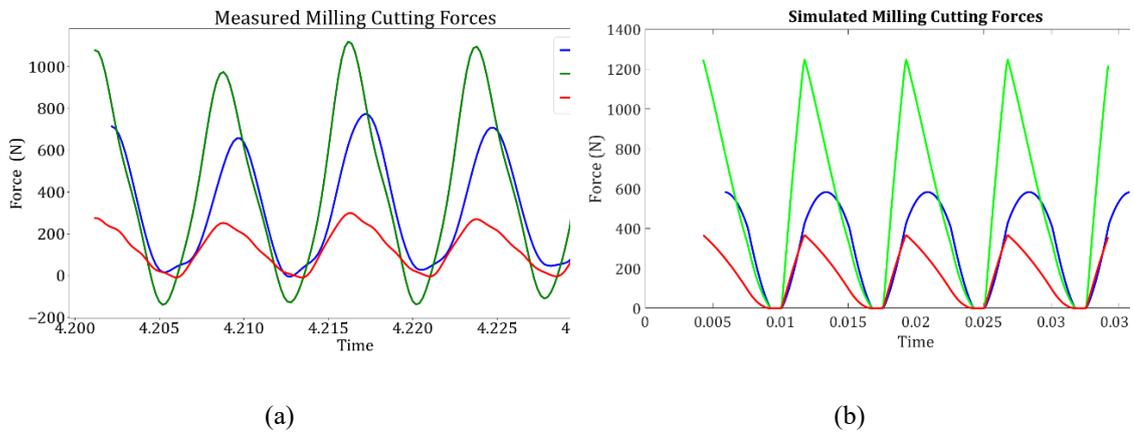


Figure 5. (a) Filtered Measured data (b) Milling Simulation

## 3.3. Results and Discussion

### 3.3.1. Cutting force prediction by hybrid ML model

Based on the mechanistic machining models as described by Equation (5), simulated cutting forces were calculated and compared with the experimental results, for three

different materials (Steel 1050, Al7075-T6 and Ti6Al4V). As summarized in Table1, the simulation accuracy lies between 86.57%-89.57% for  $F_x$ , 83.22%-87.40% for  $F_y$ , and 83.21%-89.83% for  $F_z$  during machining of various materials. The RMSE values lie between 6-10 for  $F_x$ , 7-15 for  $F_y$  and 4-6 for  $F_z$ . Due to numerous assumptions made during the milling simulations, the cutting forces predictions are not precise enough. This issue is well illustrated in Figures 6 and 7, which demonstrate the comparison between the simulated and measured cutting forces.

Table 1. Comparison of simulated cutting forces and experimental results.

Material		$F_x$ (N)	$F_y$ (N)	$F_z$ (N)
Aluminum 7075-T6	R <sup>2</sup> (%)	88.3	87.40	83.21
	RMSE	8.04	11.18	6.31
Steel 1050	R <sup>2</sup> (%)	86.57	83.22	89.07
	RMSE	10.07	15.22	6.36
Ti6Al4V	R <sup>2</sup> (%)	89.53	73.69	89.83
	RMSE	6.27	7.84	4.35

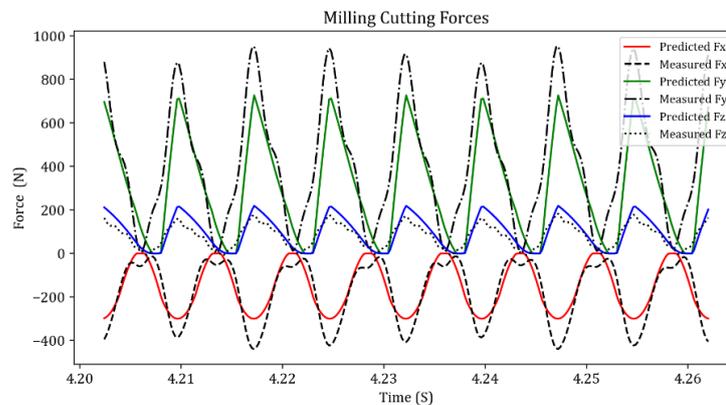


Figure 6. Predicted and measured cutting forces for AL7075-T6. (Axial depth of cut is 4mm, Radial depth of cut is 4mm, Spindle speed is 2000 rpm, Feed rate is 0.25 (mm/rev\*tooth))

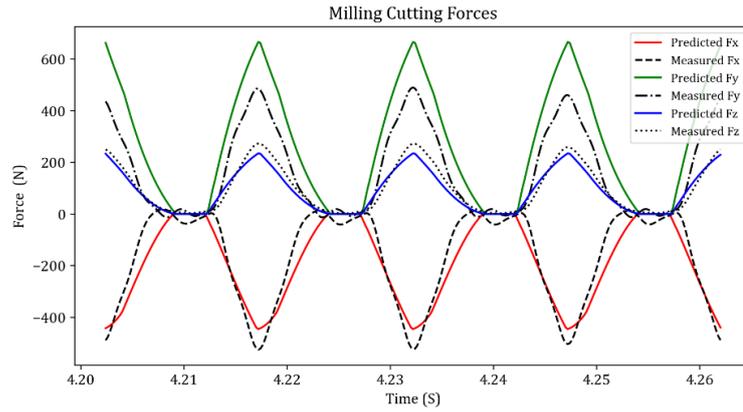


Figure 7. Predicted and measured cutting forces for Ti6Al4V (Axial depth of cut is 6mm, Radial depth of cut is 2mm, Spindle speed is 1000 rpm, Feed rate is 0.125 (mm/rev\*tooth))

Since the simulated cutting forces weren't sufficiently accurate, a hybrid physics-based ML model was developed, which was trained using both analytical and experimental data. By ensuring the generality of such a hybrid ML model, the need for expensive experimental tests can be minimized. Meanwhile, when the mechanistic machining models are improved through hybrid ML models, they can be employed with a high level of reliability to calculate the cutting forces of various materials with various cutting tools, and under a variety of milling conditions.

The hybrid physics-based ML model used in this study was trained by measurements datasets of three different materials, including Steel 1050, Aluminum 7075-T6 and Ti6Al4V. In selecting these materials, consideration was given to the fact that they are widely used across a variety of industries. Additionally, their thermo-mechanical properties cover the properties of various materials that are used extensively in machining processes. In this study, a wide range of input parameters have been used to train various machine learning algorithms. The presented algorithms were trained using a very limited number of tests, and this significantly increased the accuracy of milling simulations to 98.90%. Furthermore, the range of RMSE values narrowed, decreasing from “4-15” to the more concise range of “3-6”. The input parameters which have been used for these algorithms are radius, teeth number, helix and rake angle of the tool, axial and radial depth of cut, feed rate, spindle speed and material characteristics of the workpieces. To conduct a suitable set of experiments, the composite central design (CCD) method was utilized. One of the most significant advantages of this algorithm is reducing the number of tests

for further workpiece materials. It means that if a model is created using three different materials, the requirement for real measurements for the third material is less than that for the other two materials, and the need to conduct tests with the second material is less than that for the first workpiece material. As illustrated in Table 1 different materials have been used to conduct the proposed model. Based on the results obtained from the ML model trained only with Al7075-T6, fewer training datasets were prepared for Steel 1050 and Ti6Al4V. In other words, since the hybrid machine learning model could successfully discover the relationship between cutting forces and input parameters for aluminum, it was not necessary to perform as many tests as those conducted for aluminum on the other mentioned materials. In addition, due to the high cost and technological difficulties involved in Ti6Al4V machining, only a few experimental tests were conducted on this material. As a result, the training dataset contained only a small portion of Ti6Al4V machining data. In order to define the materials for the machine learning model, two different scenarios can be applied, i.e., quantitative and qualitative. Material types are defined qualitatively through categorical parameters such as type A, type B, etc. This type of material definition is limited to trained cases only. Therefore, to increase the flexibility and generality of the proposed model, thermo-mechanical and physical properties were utilized as quantitative material indicators in this study.

The wide range of input parameters which have been used for generating databases for training of machine learning algorithms are listed in Table 2. The tool geometries, cutting parameters, thermo-mechanical properties of workpieces and the simulated cutting forces were considered as the inputs; and the experimentally measured cutting forces as the outputs of the physics-based ML model.

Table 2. Input parameters

<b>Input Parameters</b>	<b>Levels</b>
Tool radius (mm)	12, 16 and 20
Teeth number	3 and 4
Helix angle (°)	35 and 50
Rake angle (°)	5 and 7
Runout (µm)	23 and 37
Axial depth of cut (mm)	1, 4, 11, 17 and 30
Radial depth of cut (mm)	1, 2, 4, 8

Spindle speed (rpm)	1000-2000 and 4000
Feed rate (mm / rev.teeth)	0.05, 0.15 and 0.25
Workpiece material	Aluminum 7075-T6, Steel 1050, Ti6Al4V

The regression analysis was conducted in the MATLAB<sup>®</sup> and Python environment using three different ML models, namely random forest (RF), gradient boosting (LSBoost) and support vector regression (SVR). Initially, a Bayesian optimization algorithm was employed to optimize the hyperparameters of the ML models, which are listed in Table 3. To avoid over-fitting the models, a five-fold cross validation error was used as the objective function of the Bayesian algorithm.

Table 3. Hyperparameters of ML models and their optimum values.

ML Models	Hyperparameters	F <sub>x</sub>	F <sub>y</sub>	F <sub>z</sub>
SVR	Box Constraint	982.67	943.09	992.83
	Epsilon	15.234	11.771	6.231
	Kernel Function	Gaussian	Gaussian	Gaussian
Random Forest	Minimum leaf size	3	3	1
	No. predictors to sample	13	14	7
	No. of Trees	15	15	15
	In bag Fraction	0.83193	0.7596	0.5656
LSBoost	No. of Learning Cycles	320	156	499
	Learning rate	0.1412	0.2426	0.0928
	Minimum leaf size	58	29	21
	Maximum No. of splits	18402	17944	3597
	No. of variables to sample	4	5	3

After hyperparameter optimization, the regression analysis for each machine learning algorithm was conducted. A summary of obtained results is listed in Table 4. As demonstrated, all ML models have a high coefficient of determination ( $R^2$  more than 97%) with low RMSE values (i.e., in the range of 3-6). Root mean square error (RMSE) is a widely used performance metric in machine learning to assess the accuracy of regression models. It quantifies the average magnitude of the differences between predicted and actual values. Lower RMSE values indicate better model performance, as

they signify smaller prediction errors, while higher RMSE values suggest less accurate predictions. Subsequently, the highest performance (i.e., the lowest RMSE and highest  $R^2$  values of the *unseen test* dataset) is associated with the LSBoost model, followed by SVR and RF, respectively. The SVR model exhibits significantly longer training times compared to LSBoost and RF. While LSBoost and RF demonstrate similar  $R^2$  and RMSE values, LSBoost slightly outperforms RF. The adjusted  $R^2$  values of LSBoost model are 98.27%, 99.43% and 99.09% for the training dataset; and are 97.32%, 98.9% and 98.51% for the test dataset, for the cutting forces of  $F_x$ ,  $F_y$  and  $F_z$ , respectively. The regression curves for both training and testing data set are illustrated in Figures 6-8.

Table 4. Performance parameters of the ML models for unseen test dataset.

<b>ML Models</b>	<b>Metrics</b>	<b>F<sub>x</sub></b>	<b>F<sub>y</sub></b>	<b>F<sub>z</sub></b>
SVR	RMSE	5.97	5.95	3.52
	Adj. R <sup>2</sup> (%)	97.28	98.92	98.45
LSBoost	RMSE	5.94	5.76	3.54
	Adj. R <sup>2</sup> (%)	97.32	98.9	98.51
Random Forest	RMSE	6.07	6.01	3.62
	Adj. R <sup>2</sup> (%)	97.09	98.89	98.29

Table 4 illustrates the performance of various machine learning algorithms when applied for simulation improvements. The LSBoost algorithm demonstrates the best performance compared to the other two methods. As shown in the table, the performance of improved milling process prediction has increased to 98.9%. Such a high prediction accuracy can reduce the need for costly and time-consuming measurements resulting in a better understanding of the process and higher quality of products. Figure 8 shows the correlation of both training and test data set.

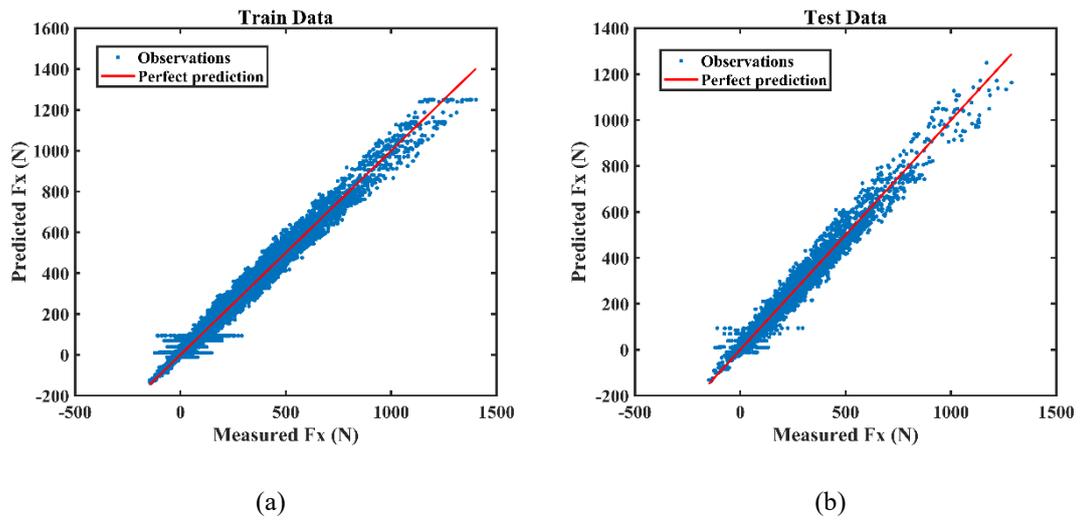


Figure 8. Regression curves of LSBoost model for cutting forces in X direction:  
a) Training data, b) Test data

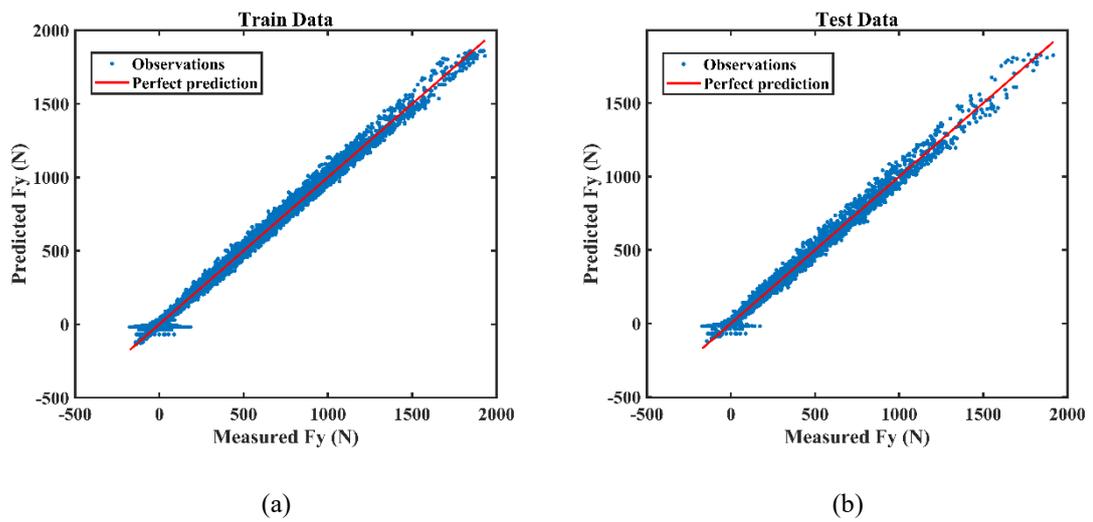


Figure 9. Regression curves of LSBoost model for cutting forces in Y direction:  
a) Training data, b) Test data

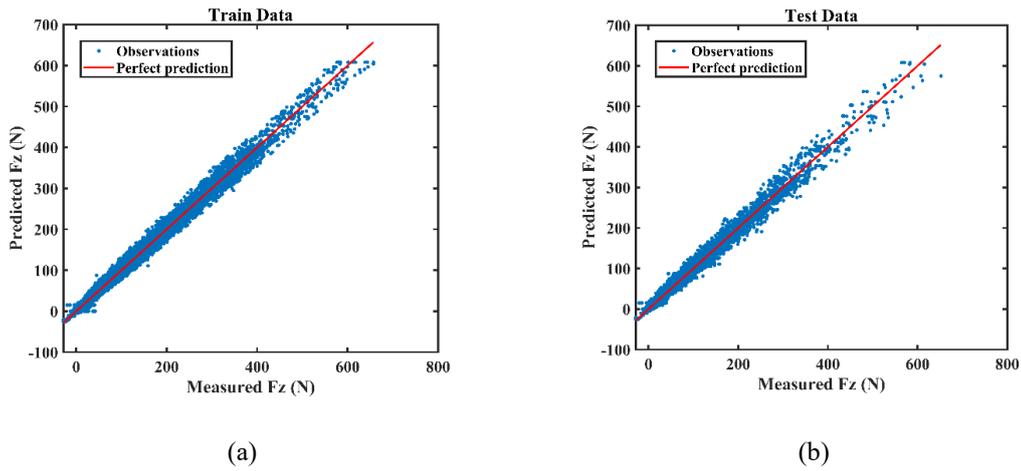
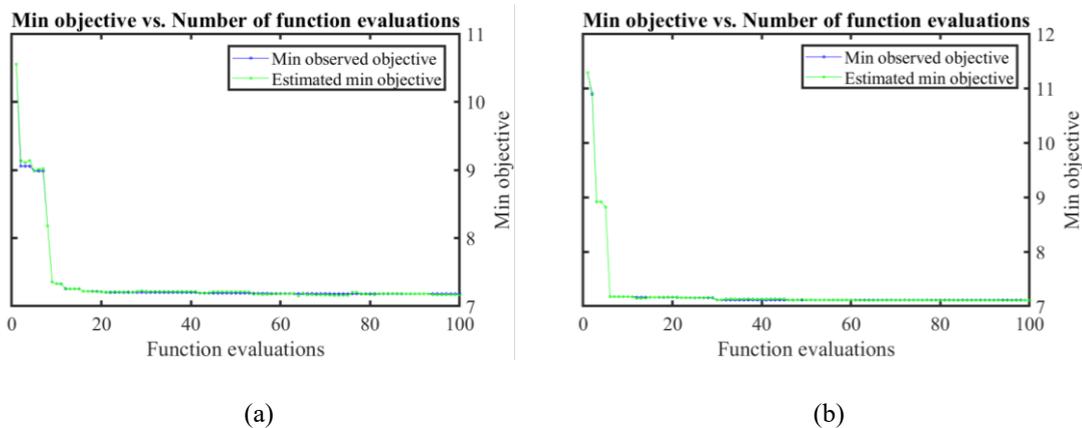
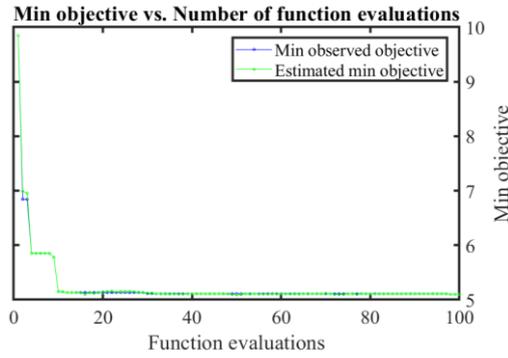


Figure 10. Regression curves of LSBoost model for cutting forces in Z direction:

a) Training data, b) Test data

Moreover, the evolution of the optimization process for LSBoost model is illustrated in Figure 11. As shown in the mentioned figure, one of the most important aspects is the convergence observed throughout the optimization process by utilizing Bayesian algorithms. As the number of iterations increases, the algorithm progressively narrows its search, and the points representing evaluated hyperparameters steadily gravitate towards a specific region of the hyperparameter space. This convergence signals a compelling ability of the algorithm to pinpoint optimal configurations, resulting in the minimization of the objective function. Min observed objective represents the objective function the lowest value of observed during the hyperparameter optimization process and estimated min objective is an approximation of the true minimum value of the objective function, obtained through optimization algorithms.



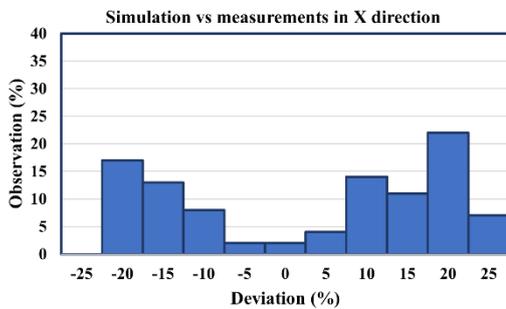


(c)

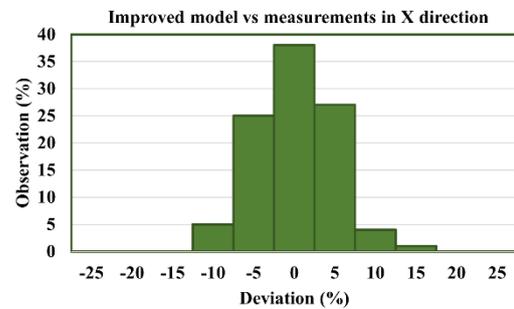
Figure 11. Evolution of hyperparameter optimization process for LSBoost model:

a)  $F_x$ , b)  $F_y$  and c)  $F_z$ .

A statistical error analysis was conducted to illustrate the accuracy of the proposed ML model. To evaluate the performance of the simulation and the improved ML model in comparison with experimental cutting forces, fifty test data sets were selected. These experiments were performed by using different axial (1, 2, 3 mm), radial depth of cut (2, 4, 8 mm), feed (0.05, 0.1, 0.15 mm/rev.tooth), spindle speed (1000, 3000 rpm) and various materials (i.e., Al7075-T6, Steel 1050 and Ti6Al4V). In this regard, for each test setup the maximum cutting force was selected as the performance metric. The improved simulation model shows a significant reduction in deviation for the maximum cutting force (see Figure12). Furthermore, the error distribution was remarkably tightened to a very narrow range. The majority of deviations in the improved model were within a narrow band of -5 to 5%, representing 90% of the observations. According to this error analysis, the hybrid model provides superior performance in predicting the milling cutting forces.



(a)



(b)

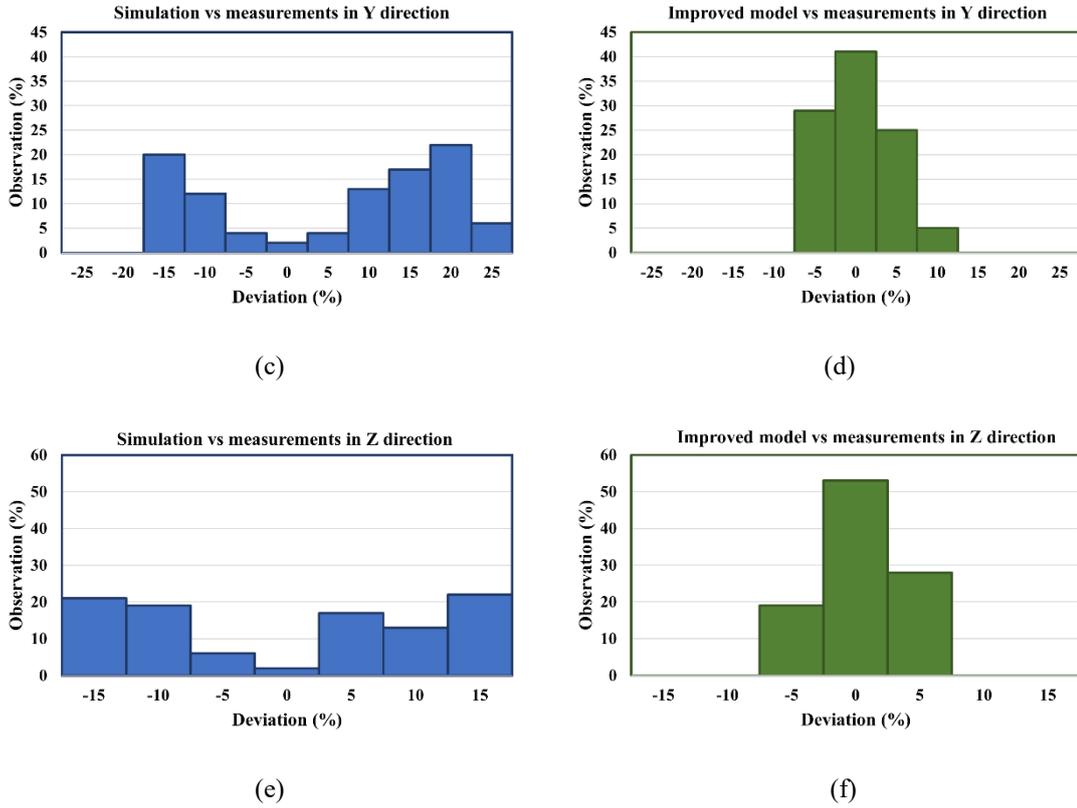


Figure 12. Statistical error analyses for simulation and hybrid model

### 3.3.2. Prediction accuracy of unseen material

The physics-based machine learning models proposed in this study demonstrated high accuracy in predicting cutting forces for a range of materials, such as Aluminum 7075-T6, Steel 1050, and Ti6Al4V. As previously mentioned, these materials were characterized based on their thermo-mechanical properties. Consequently, the model exhibits the capability to predict cutting forces for materials not encountered during training. To assess both accuracy and generality, the proposed hybrid model was utilized to predict cutting forces for the unseen material Inconel 625 superalloy. As depicted in Table 5, the RMSE values are 5.47, 6.15 and 3.51 for  $F_x$ ,  $F_y$  and  $F_z$ , respectively. The adjusted  $R^2$  values are 96.43%, 95.76% and 97.02% for  $F_x$ ,  $F_y$  and  $F_z$  respectively, which indicates that the developed physics-based ML model could predict the cutting forces of completely different materials by using the properties of the three trained materials. Indeed, by transferring the acquired knowledge of the trained ML model on St1050, Al7075-T6 and Ti6Al4V, the cutting forces of another material (i.e., Inconel 625) were

predicted with a high level of accuracy, which proves the model’s generality. In other words, incorporating the analytical models into the ML algorithms enabled the trained model to discover the relationship between the input features and the output responses very accurately, i.e., it could well capture the dependency of the cutting forces on the “machining parameters, cutting conditions and thermo-mechanical properties”. Consequently, such a hybrid model can predict the cutting forces of a different material, under a variety of cutting conditions. Figure12 Shows the machine learning based predicted cutting forces of Inconel 625. To conduct this measurement, the tool radius is 16 mm, tooth number of the tool is 4, helix angle is 35°, rake angle is 5°, spindle speed is 1000 (rpm), feed rate is 350 (mm/min), axial depth of cut is 4 (mm), radial depth of cut is 1 (mm) and the milling process is down-milling. As illustrated in Figure12, there is a huge improvement in prediction accuracy when machine learning methods have been applied. In this prediction the aforementioned ML algorithm has been trained by using three specified materials (i.e., Steel 1050, Aluminum 7075-T6 and Ti6Al4V). Moreover, the trained algorithm has been used for predicting the cutting forces of Inconel 625 by only importing the thermo-mechanical properties of this material. Figure 13 illustrates the prediction accuracy of the utilized method.

Table 5. Performance metrics for unseen dataset of Inconel 625

ML Model	Metrics	F <sub>X</sub>	F <sub>Y</sub>	F <sub>Z</sub>
LSBoost	RMSE	5.47	6.15	3.51
	Adj. R <sup>2</sup> (%)	96.43	95.76	97.02

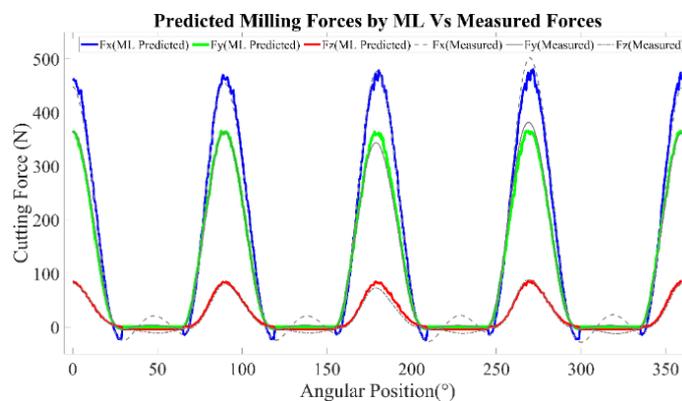


Figure 13 Machine learning based prediction. of Inconel 625 cutting forces

The thermo-mechanical properties of the materials utilized in training and prediction of machine learning algorithm have been illustrated in Table 6.

Table 6. Thermo-mechanical properties of work-piece materials

Material	Yield Strength (MPa)	Young's Modulus (GPa)	Hardness (HB)	Thermal Conductivity (W/mK)	Thermal Expansion Coef. (1/°C) *e <sup>-6</sup>	Density (Kg/m <sup>3</sup> )	Ultimate Tensile Strength (MPa)
ST1050	515	200	182	51.9	14.7	7.87	620
AL7075-T6	503	72	150	130	25.2	2.81	572
Ti6Al4V	830	115	340	6	9	4.706	900
Inconel 625	517	207	190	9.8	12.8	8.44	930

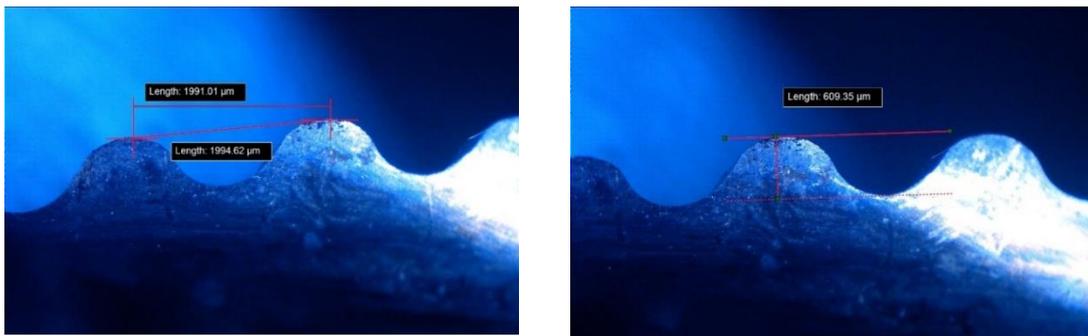
### 3.3.3. Prediction accuracy of special milling tools

A complementary hybrid ML model was developed to predict the cutting forces during milling with various cutting tool geometries, such as: tapered ball end mills, serrated end mills and tapered serrated ball end mills. The geometrical specifications of the tools have been shown in Table7. In this regard, some experimental tests were carried out with these cutting tools on Aluminum 7075-T6; meanwhile the simulated cutting forces were estimated using mechanistic machining models, as described by Equation (5). Then the training data set was prepared, containing the machining parameters, cutting tool geometries, and simulated cutting forces as the inputs, and the experimental cutting forces as the targets. Moreover, this data has been added to the database that was prepared for normal endmills. In normal tools, the geometric aspects related to serration geometry, such as wavelengths and amplitude, were considered as zero. These values for special milling tools are illustrated in Table7. The predicted cutting forces for the aforementioned cutting tools are depicted in Figure15, and the RMSE and adjusted  $R^2$  values are summarized in Table8. The RMSE values for the unseen test data lie between 2.85 and 6.18, with  $R^2$  values of 98.76%, 98.17%, and 98.62% for  $F_x$ ,  $F_y$ , and  $F_z$ , respectively, indicating that the hybrid physics-based ML model could accurately predict the cutting forces involved in the milling process with various cutting tools geometries. The obtained

results reveal that the proposed approach in this research not only helped minimize the required experimental tests for training the ML model, but also improved the accuracy of the mechanistic model, making it more reliable in real-world applications, for various materials, cutting tools, machining parameters and conditions.

Table 7. Special tools geometrical specifications

<b>Tools</b>	<b>Geometrical specs</b>
<b>Tapered ball end mills</b>	Nose ball radii = 3 mm, Root radii = 13.5 mm, Tapered angle = 9°
<b>Serrated end mills</b>	Radius = 6 mm, Wavelength = 2 mm, Amplitude = 0.3 mm
<b>Tapered serrated ball end mills</b>	Nose ball radii = 2.5 mm, Root radii = 7 mm, Wavelength = 4.5 mm, Amplitude = 0.350 mm



a)

(b)

Figure 14. Serrated endmills (a)Wavelength (b)Amplitude measurements.

Table 8. Performance metrics for special milling tools

<b>ML Models</b>	<b>Metrics</b>	<b>F<sub>X</sub></b>	<b>F<sub>Y</sub></b>	<b>F<sub>Z</sub></b>
SVR	RMSE	5.76	6.17	3.17
	Adj. R <sup>2</sup> (%)	96.24	95.81	97.73
LSBoost	RMSE	4.97	5.95	2.85
	Adj. R <sup>2</sup> (%)	98.76	98.17	98.62

Random Forest	RMSE	5.48	6.18	3.35
	Adj. R <sup>2</sup> (%)	97.94	95.37	96.24

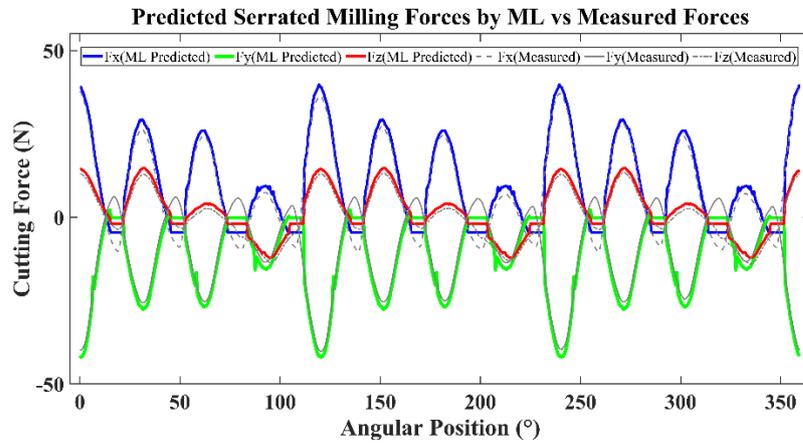


Figure 15. Comparison of serrated tools cutting forces predicted by ML and experimental tests.

### 3.3.4. Models' performance for out-of-range input values

In this study, the machine learning models were trained with nine input parameters, each defined within an acceptable wide range (see Table 2), which is more common in industrial applications. The range of input parameters for training datasets can be defined by using historical data in manufacturing lines. In this section of study, the performance of developed methods in the cases of input parameters are out of trained range has been investigated. As mentioned previously, these algorithms have very high accuracy when all test input parameters are in the range of training data sets (i.e., 98.9%). However, the performance of an ML model should be evaluated in predicting the cutting forces for out-of-range input values. For this purpose, several datasets were prepared, each one having one or more out-of-range input parameters. In this respect, this procedure was applied for all the input parameters (i.e., tool radius, teeth number, rake and helix angle, feed rate, spindle speed, axial and radial depth of cut and runout). Four sets of tests produced by the combination of  $n$  ( $n=1\sim4$ ) parameters out of nine parameters have been considered and the average performance of model for each of the sets is found. According to the obtained results, when a single input parameter deviates from its training range, the model

maintains high accuracy, achieving 96.5% average accuracy. In cases where two input parameters fall out of range simultaneously, the model's accuracy decreases to 91.7%. Further deviations occur when three input parameters are out of range, leading to a reduced accuracy of 85%. The most significant accuracy drop is observed when four input parameters are simultaneously out of their training ranges, resulting in 72% and in this case the simulation results have better accuracy (see Figure16). For instance, one of the previously mentioned test setups aimed to assess the accuracy of the proposed algorithm involves investigating out-of-range input parameters, such as axial depth, radial depth, feed rate, and spindle speed. In conducting these tests, each parameter has been employed in combinations of both in-range and out-of-range values. In cases where the axial depth of cut is out of range (OR) while other parameters are in range (IR), the accuracy of prediction exceeds 96% compared to the measured data. In the second scenario, combining out-of-range axial and radial depths with in-range parameters results in an accuracy of over 91%. However, in tests involving 4 out-of-range inputs, axial, radial depth of cut, feed rate, and spindle speed, the model's accuracy decreases to almost 70%.

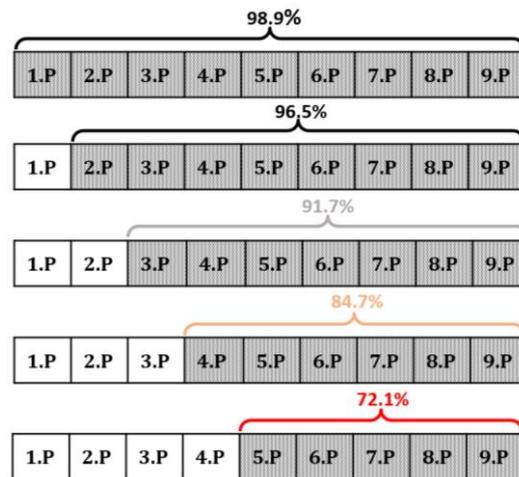


Figure 16. The prediction accuracy for out-of-range input.

### 3.4. Summary

In this study a physics-based machine learning model was developed to improve the accuracy of analytical prediction of the cutting forces during the milling process of various materials with different cutting tools and under a variety of cutting conditions. In addition, the present study comprehensively investigated the feasibility of the hybrid

model in decreasing the number of experimental tests required for preparing training dataset. For this purpose, three different ML algorithms (i.e., SVR, RF and LSBoost) were employed, which were trained by utilizing both analytical models and experimental data. The following conclusions were drawn from the present study:

- The simulation accuracy of the mechanistic milling models (without implementing ML) for various materials (i.e., Steel 1050, Aluminum 7075-T6 and Ti6Al4V) was between 86.57%-89.46% for  $F_x$ , 83.22%-87.40% for  $F_y$ , and 83.21%-89.83% for  $F_z$ . Nevertheless, these values suggest that significant enhancements are necessary to make the simulation models more practical in real-world industrial applications.

- By developing a hybrid model based on both analytical models and experimental data, high prediction accuracy (more than 97%) was obtained, even for an unseen dataset of completely different conditions. Comparison of the obtained results indicates a 14% improvement in cutting force predictions by hybrid model.

- Among various ML models, the best performance was achieved by the LSBoost model, followed by support vector regression and random forest, respectively.

- Incorporating the mechanistic milling models into the ML model enabled it to discover the complex relationship between the cutting forces and machining parameters, cutting conditions and thermo-mechanical properties.

- Since the materials were characterized using their thermo-mechanical and physical properties, by transferring the acquired knowledge of the ML model for Steel 1050, Aluminum 7075-T6 and Ti6Al4V, the cutting forces of unseen material (Inconel 625) were also predicted with a high level of accuracy (i.e., 95%), proving the model's generality.

- Furthermore, a complementary hybrid ML model was proposed to predict the milling forces of various special tool types including end mills with normal and serrated edges with different shapes (i.e., cylindrical, and tapered end mills with flat, ball and round noses). The developed hybrid ML model could also accurately predict the cutting forces of these tools, where the coefficient of determination was more than 98%.

- The performance of the proposed model was evaluated in terms of predicting the cutting forces for the out-of-range input parameters. In this respect, several datasets were

prepared, each one having one or more out-of-range input parameters. The obtained results indicated that the prediction accuracy of the ML model was remarkably high even in the case of two simultaneous out-of-range input parameters.

- This approach offers a significant benefit wherein the machine learning database can be enhanced through the continuous collection of data during the manufacturing process and on the production line, leveraging Internet of Things (IoT) and Industry 4.0 principles.
- The proposed physics-based approach not only helped minimize the required experimental tests for training the ML model, but also improved the accuracy of the mechanistic model, making it more reliable for real-world applications.

### **3.5. Future work**

The obtained results provided valuable insights into the application of hybrid machine learning models in accurately predicting the cutting forces of various materials, tool types, and machining conditions. The significance of our findings lies in their potential implications for real-time condition monitoring systems. Therefore, such hybrid models can be used for condition monitoring of machining processes, where a high level of reliability and accuracy is required. Furthermore, the proposed model not only addresses current challenges but also serves as a robust foundation for process optimization purposes. Considering this foundational aspect, the model can be viewed as the basis for industrial quality improvement.

## 4. SMART TOOL-RELATED FAULTS MONITORING SYSTEM USING PROCESS SIMULATION-BASED MACHINE LEARNING ALGORITHMS

### 4.1. Milling Force Model

In the present study, the linear edge to calculate milling forces for end-milling tools and these simulations were calibrated using measured test force model [29] is employed data and machine learning algorithms. In order to this input parameters and results of simulation are used as inputs for machine learning algorithms and the outputs of machine learning algorithms were measured data. By using this method, the accuracy of analytical simulations has been improved. To determine cutting forces for each angular increment of the tool, differential forces are computed for every axial element ( $i$ ) on each tooth ( $j$ ) at a specific rotational position ( $\phi$ ) throughout a complete rotation of the cutting tool:

$$\begin{aligned}
 \varphi_{i,j}(\phi) &= \varphi_{i,j} + \phi \\
 dF_r(i,j,\phi) &= g(\varphi_{i,j}(\phi)) [K_{re} + K_{rc}(i,j)h_{i,j}(\phi)]dz \\
 dF_t(i,j,\phi) &= g(\varphi_{i,j}(\phi)) [K_{te} + K_{tc}(i,j)h_{i,j}(\phi)]dz \\
 dF_a(i,j,\phi) &= g(\varphi_{i,j}(\phi)) [K_{ac}(i,j)h_{i,j}(\phi)]dz
 \end{aligned} \tag{6}$$

Where the cutting force coefficients  $K_{rc}$ ,  $K_{tc}$  and  $K_{ac}$  are calculated using the oblique cutting force model, combined with orthogonal cutting data [25], while taking into account local oblique angles ( $\eta_{i,j}$ ) for each element. The edge force coefficients  $K_{re}$ ,  $K_{te}$  and  $K_{ae}$  are typically determined from cutting tests, but can also be predicted using thermo-mechanical models applied to the third deformation zone [120]. In calculating the force coefficients, the rake angle on the cutting edges is assumed to be constant; however, it may vary along the cutting edges depending on the manufacturing process of these tools. In such cases, the local rake angle should be utilized in force coefficient calculations [118,119].

The binary function  $g(\varphi_{i,j}(\phi))$  equals 1 when the element is in cut (i.e.  $\varphi_{start} \leq \varphi_{i,j}(\phi) \leq \varphi_{exit}$ ) and 0 otherwise.  $\varphi_{i,j}(\phi)$  represents the angular position of each point on the edge when the tool's rotation angle is  $\phi$ .  $dz$  denotes the thickness of each axial element. As depicted in Figure 17,  $\Delta\varphi_{i,j}$  differs for each edge at a specific axial position,

and therefore,  $h_{i,j}(\phi)$  (chip thickness) can be defined as follows:

$$h_{i,j}(\phi) = \frac{\Delta\varphi_{i,j}}{2\pi} Nf \sin(\varphi_{i,j}(\phi)) \quad (7)$$

where  $N$  represents the number of teeth and  $f$  corresponds to the nominal feed per tooth.

The total forces in  $x, y, z$  directions for angular orientation of the tool can be obtained by summation of the elemental differential forces:

$$F_x(\phi) = \sum_{i=0}^a \sum_{j=1}^{N_t} \left[ -dF_r(i, j, \phi) \sin(\varphi_{i,j}(\phi)) - dF_t(i, j, \phi) \cos(\varphi_{i,j}(\phi)) \right]$$

$$F_y(\phi) = \sum_{i=0}^a \sum_{j=1}^{N_t} \left[ -dF_r(i, j, \phi) \cos(\varphi_{i,j}(\phi)) + dF_t(i, j, \phi) \sin(\varphi_{i,j}(\phi)) \right] \quad (8)$$

$$F_z(\phi) = \sum_{i=0}^a \sum_{j=1}^{N_t} dF_a(i, j, \phi)$$

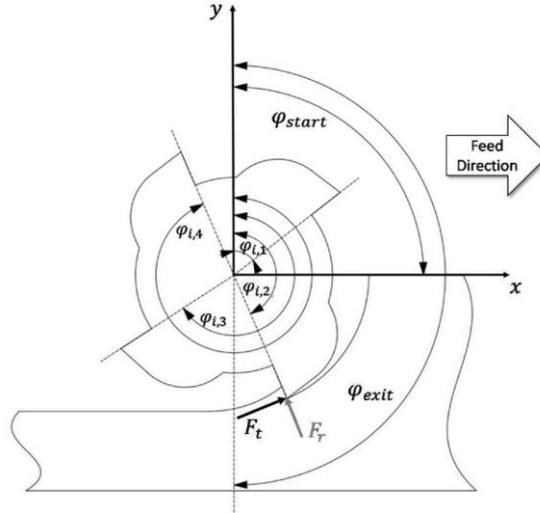


Figure 17. The schematic view of the milling cutting force directions [121]

#### 4.2. Applied Machine Learning for Fault Detection

Since fault diagnosis is the most challenging aspect of process and machine repair, the majority of downtime is spent localizing the fault rather than addressing it [122]. Consequently, organizations are exploring innovative methods to enhance the root cause

analysis (RCA) process for faults.

As a first step, it is essential to define the problem and determine the appropriate data analytics techniques. In order to make the required data suitable for further analysis, it is necessary to collect, and preprocess the data based on the problem and the selected method. As a result, developing, and evaluating a data model is vital. To resolve the issue, the outcomes are examined. The process is typically repeated several times in order to achieve better results.

Based on the type of input data and the type of learning system, machine learning algorithms can be divided into three categories. In supervised learning, algorithms are trained to map inputs to known outputs (provided by experts). In unsupervised learning, models or functions are developed without incorporating any previously known outputs. This approach typically analyzes large datasets to discover meaningful patterns or classifications. Additionally, reinforcement learning allows a machine to determine its performance based on a reward signal that has been previously defined.

Two major objectives of these algorithms are to classify or cluster data, and to identify a trend or relationship over time, respectively. This study investigated a number of machine learning algorithms for data training and fault detection and selected three algorithms based on the results: Multiple Linear Regression, K-Nearest Neighbor (KNN), and Random Forest.

**Multiple Linear Regression (MLR):** Multiple linear regression is a statistical method used to model the relationship between a dependent variable and two or more independent variables. It is an extension of simple linear regression, which involves only one independent variable. The primary goal of multiple linear regression is to create a predictive model that can estimate the value of the dependent variable based on the values of the independent variables. The multiple linear regression model takes the form:

$$y = \beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + \dots + \beta_n * x_n + \varepsilon \quad (9)$$

Here,  $y$  represents the dependent variable,  $x_1, x_2, \dots, x_n$  are the independent variables,  $\beta_0$  is the intercept,  $\beta_1, \beta_2, \dots, \beta_n$  are the regression coefficients, and  $\varepsilon$  is the residual or error term, which accounts for the variation in the data not explained by the model.

The regression coefficients ( $\beta_1, \beta_2, \dots, \beta_n$ ) represent the change in the dependent variable associated with a one-unit change in the corresponding independent variable, while holding all other variables constant. These coefficients are estimated using a technique called ordinary least squares (OLS), which minimizes the sum of the squared differences

between the observed values of the dependent variable and the values predicted by the model.

**K-Nearest Neighbor (KNN):** K-Nearest Neighbor (KNN) is a non-parametric, instance-based, supervised learning algorithm used for classification and regression tasks. It is considered one of the simplest and most intuitive machine learning algorithms, owing to its easy-to-understand approach and minimal training requirements. In KNN, the prediction for a new data point is determined based on the  $K$  closest data points (neighbors) from the training dataset. The algorithm operates under the assumption that similar data points are located near each other in the feature space. Euclidean distance is a commonly used distance metric in the KNN algorithm. It is the straight-line distance between two points in an  $n$ -dimensional space [123,124]. The algorithm calculates the Euclidean distance between the new data point and every point in the training dataset, then sorts these distances in ascending order.

The choice of  $K$  is crucial in KNN, as it directly affects the algorithm's performance. A small value of  $K$  might result in overfitting, while a large value of  $K$  can lead to underfitting. Typically, the optimal value of  $K$  is determined through techniques like cross-validation. KNN is sensitive to the scale of the input features, so it is often necessary to normalize or standardize the data before applying the algorithm. Furthermore, KNN is sensitive to the presence of irrelevant or noisy features, which can negatively impact its performance. Feature selection techniques can help mitigate this issue.

Below is a simple pseudocode for the K-Nearest Neighbor (KNN) algorithm for classification:

- Function KNN\_Classify (new\_data\_point, training\_data,  $K$ ):
- Initialize an empty list called `distances_list`.
- For each `data_point` in `training_data`:
- Calculate the distance (e.g., Euclidean distance) between `new_data_point` and `data_point`.
- Add (`distance`, `data_point`'s class) to the `distances_list`.
- End For
- Sort `distances_list` in ascending order of distance
- Select the first  $K$  elements from the sorted `distances_list`.
- Initialize an empty dictionary called `class_votes`.
- For each (`distance`, `class`) in the  $K$  selected elements:
- If `class` is not in `class_votes`:
- `class_votes[class] = 0`

- End If
- class\_votes[class] += 1
- End For
- Determine the class with the highest vote in class\_votes.
- Return the class with the highest vote as the prediction for new\_data\_point.

Random Forest (RF): Random Forest is an ensemble learning method used for both classification and regression tasks. It operates by constructing multiple decision trees during the training phase and then aggregating their outputs to make a final prediction. The main idea behind the Random Forest algorithm is to combine the results of multiple weak learners (decision trees) to obtain a more accurate and robust model. It is particularly effective for handling high-dimensional data and can address classification and regression problems. Here's the logic behind how Random Forest works:

1. Bootstrapping: For a given dataset, Random Forest creates multiple bootstrap samples by randomly selecting data points with replacement. Each bootstrap sample is used to train a separate decision tree.
2. Feature Randomness: During the process of growing individual decision trees, at each node, a random subset of features is selected to determine the best split. This random feature selection introduces diversity among the trees and reduces the correlation between them.
3. Decision Tree Construction: A decision tree is built for each bootstrap sample using the selected features at each node. The tree construction continues until a maximum depth is reached or a minimum number of samples per leaf is obtained.
4. Aggregating Predictions: Once all decision trees are constructed, the Random Forest algorithm makes a prediction by combining the predictions of all trees. For classification problems, this is typically done by taking the majority vote among the tree predictions. For regression problems, the average prediction of all trees is used.

In addition to reducing overfitting risk, the Random Forest algorithm also improves model generalization by combining predictions from multiple decision trees. Furthermore, this algorithm is robust to noise and can handle a large dataset, making it an excellent choice for a variety of machine learning applications.

Here's a pseudocode representation of the Random Forest algorithm:

Procedure Random\_Forest (training\_data, num\_trees, max\_depth, min\_samples\_leaf, max\_features):

- Initialize an empty list called forest.
- For i = 1 to num\_trees:
  - a. Bootstrap\_sample = Create\_Bootstrap\_Sample(training\_data)
  - b. Tree = Build\_Decision\_Tree (Bootstrap\_sample, max\_depth, min\_samples\_leaf, max\_features)
  - c. Add Tree to forest.
- Return forest.

Procedure Create\_Bootstrap\_Sample(data):

- Randomly select data points with replacement from the given data.
- Return the created bootstrap sample.

Procedure Build\_Decision\_Tree (data, max\_depth, min\_samples\_leaf, max\_features):

- If max\_depth is reached or the number of samples in data is less than or equal to min\_samples\_leaf:
  - a. Return a leaf node with the majority class (classification) or average value (regression) of data.
- Randomly select a subset of features up to max\_features.
- Determine the best split using the selected features.
- Split the data into left and right subsets based on the best split.
- left\_child = Build\_Decision\_Tree (left\_subset, max\_depth, min\_samples\_leaf, max\_features)
- right\_child = Build\_Decision\_Tree (right\_subset, max\_depth, min\_samples\_leaf, max\_features)
- Return a decision node with the best split and left\_child, right\_child as its children.

Procedure Predict (forest, test\_data):

- Initialize an empty list called predictions.
- For each test\_point in test\_data:
  - a. tree\_predictions = [tree.predict(test\_point) for tree in forest]
  - b. prediction = Majority\_Vote (tree\_predictions) (classification) or Mean(tree\_predictions) (regression)
  - c. Add prediction to predictions.
- Return predictions.

#### **4.2.1. Dataset for Training Machine Learning Algorithms**

As a result of the application of an algorithm based on process simulation, significant advancements have been made in the monitoring of tool condition. Tool-related faults can be detected more effectively and economically by eliminating extensive laboratory testing. As a result of this research and development, this approach can help tool condition monitoring to be more practical and enhance manufacturing processes in general. As mentioned before, in order to predict by machine learning, the algorithms must train by the acceptable portion of datasets. A common practice is to split the dataset into the following portions:

1. Training set: This subset is used for training the machine learning model. The model learns the patterns and relationships within the data. A typical proportion for the training set is around 70%-80% of the entire dataset.
2. Test set: This subset is used to evaluate the performance of the trained model. It is crucial that the test set is separate from the training set and is not used during the training process. The test set typically comprises 20%-30% of the entire dataset.

#### **4.2.2. Inputs and Outputs of Trained Dataset Parameters**

To perform a milling simulation, several inputs are required to represent the milling process accurately. Some of the essential inputs include:

- Tool geometry: This includes the tool's radius, number of teeth, helix angle, rake angle, and cutting-edge geometry. These parameters are crucial for simulating the tool's interaction with the workpiece.
- Workpiece material: The material properties of the workpiece and cutting force coefficients during the milling process.
- Cutting conditions: These include cutting parameters such as spindle speed, feed rate, and depth of cut, which affect the cutting forces.
- Cutting tool material: The cutting tool material affects the tool's performance, wear resistance, and cutting forces. Common cutting tool materials include high-speed steel (HSS), carbide, and polycrystalline diamond (PCD).
- Coolant and lubrication: The type and application method of coolant or lubricant used in the milling process impact the tool's temperature, cutting forces, and chip formation.

The input parameters employed for constructing the database in this study are as follows:

- Tool diameter: 10, 14, 18 and 20. (mm)
- Teeth number: 3,4, and 6
- Tool helix angle: 30, 35, and 45°
- Tool rake angle: 5, 7 and 11°
- Spindle speed: 4000, 6000 and 8000 (rpm)
- Feed per revolution per tooth: 0.1, 0.3, 0.5, 0.7, 0.9 and 1. mm/(rev\*tooth)
- Axial depth of cut: 1, 3, 5, 7, 9 and 10. (mm)
- Radial depth of cut: 0.4, 1.6, 2, 2.4, 2.8, 3.6 and 4 (mm)

Utilizing these input parameters, a total of 81,648 simulations were conducted.

This study evaluated a number of machine learning algorithms. Root-mean square error (RMSE) and mean absolute error (MAE) are two commonly used metrics for evaluating such algorithms.

- Root Mean Squared Error (RMSE):

RMSE is a measure of how well the machine learning model fits the data, similar to MSE. However, RMSE is the square root of the average squared difference between the predicted values and the actual values, which makes it more interpretable in the same units as the target variable. The equation for RMSE is:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^n (y_{\text{pred}} - y_{\text{actual}})^2}{n}} \quad (10)$$

Where  $n$  is the number of observations in the test set,  $y_{\text{pred}}$  is the predicted value of the target variable and  $y_{\text{actual}}$  is the actual value of the target variable. A lower RMSE indicates that the model is better at predicting the target variable. However, it is also sensitive to outliers in the data.

- Mean Absolute Error (MAE):

MAE is another measure of how well the machine learning model fits the data. It is the average absolute difference between the predicted values and the actual values. The formula for MAE is:

$$\text{MAE} = \frac{\sum_{i=1}^n |y_{\text{pred}} - y_{\text{actual}}|}{n} \quad (11)$$

MAE is less sensitive to outliers than RMSE because it does not square the differences between the predicted values and the actual values. However, it may not be as interpretable as RMSE because it is not in the same units as the target variable.

Table 9. Comparison of various ML algorithms

	<b>Random Forest</b>	<b>Random Tree</b>	<b>KNN</b>	<b>MLR</b>
<b>RMSE</b>	0.1070	0.1852	0.1078	1.4068
<b>MAE</b>	0.0426	0.0313	0.0462	1.1891

As previously discussed, the training-data utilized for the development of the machine learning models in this study consisted of 65% of the simulation data, while the remaining 35% was reserved as test-data. As depicted in Table 1, the Random Forest algorithm demonstrated superior performance compared to other machine learning methods. Consequently, this algorithm will be employed as the primary fault detection mechanism in the process.

### 4.3. Results and Discussion

By using machine learning algorithms, this method can gain several key insights into the detection of tool-related faults in milling processes. As described above the Randomforest algorithm has been used in order to detect tool-related faults in milling processes. It has been tested 50 different scenarios for each input parameter in order to determine whether or not this method is accurate for vast range of input parameters. Approximately 50 simulations were conducted in order to determine how well the algorithm identified faults associated with the tool radius, as an example. These simulations involved deliberately changing the radius, teeth number, helix and rake angle of the tool in the simulation, as well as executing machine learning models in order to test how well the method would perform for varying parameters related to the tool. The results of this method are presented in Figure 18:

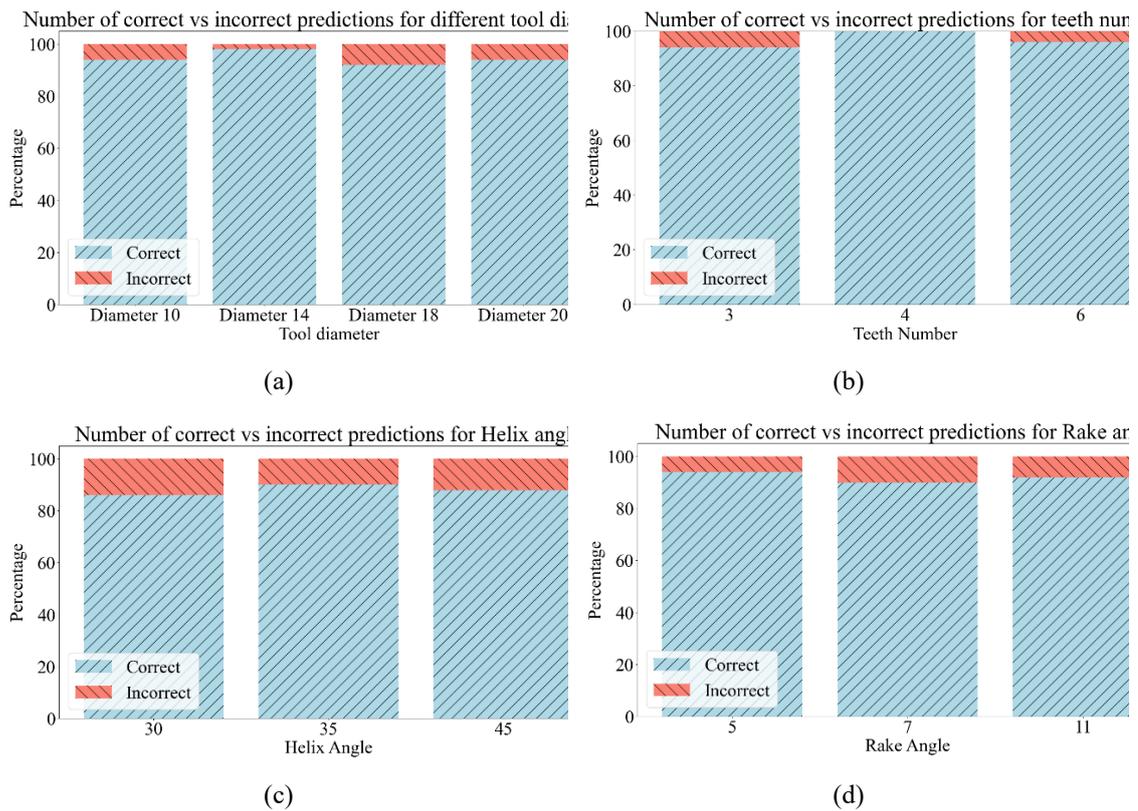


Figure 18. Correct vs incorrect fault detection of (a) Tool diameter, (b) Teeth number, (c) Tool helix angle and (d) Tool rake angle

Figure 18. illustrates the percentage of correct and incorrect prediction by using mentioned algorithms in various situations. For instance, Figure 18a. first column shows the percentage of correct prediction of tool diameter which was 10(mm). Moreover, Figure 18a. The first column shows that the algorithm can predict the cutting tool diameter with 10(mm) can be predicted with 96% accuracy. Each column of every graph shows the correct prediction percentage of that parameter. It is evident from Figure 18 that the tool parameters that have an enormous impact on the cutting force have been identified with higher precision.

The figure above indicates that the number of incorrect predictions for helix and rake angle is higher than for the other two parameters (e.g. tool diameter and tooth number). This is because helix and rake angle appear to have less impact on cutting forces compared to the tool diameter and tooth number. Aside from that, all of the tool related faults detected within acceptable precision.

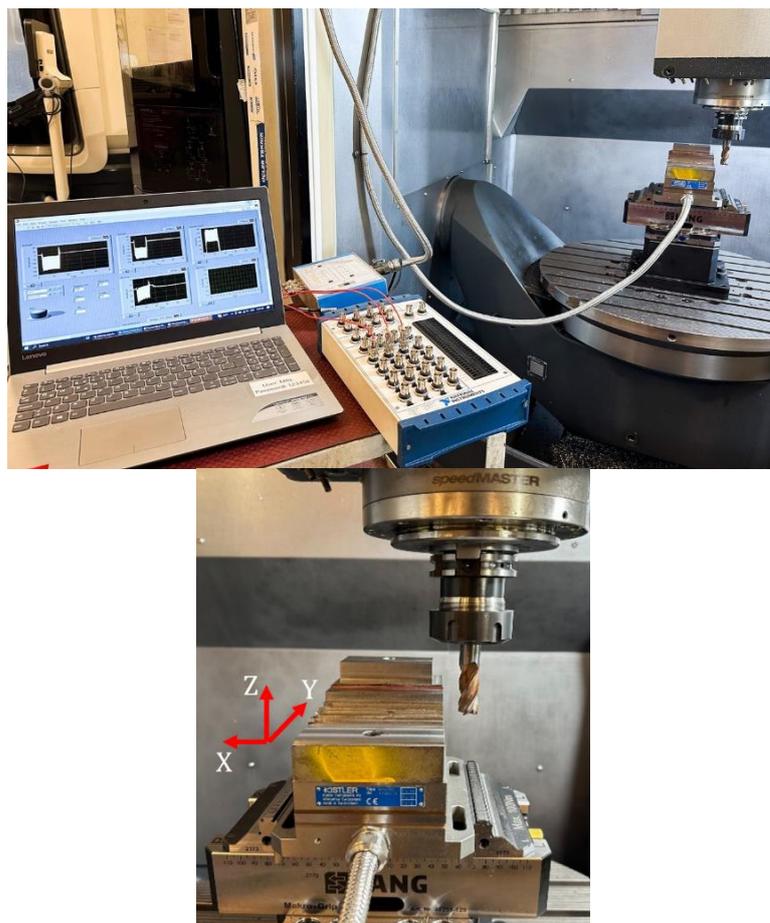


Figure 19. Measurement setup

The cutting forces were conducted using Piezo-electric Dynamometer 9257BA, amplifier and data acquisition NI USB-6259. 16 and 20 (mm) Solid Carbide end mill cutter with 4 flutes and 7075-T6 aluminum and 1050 steel were used as workpieces during the tests.

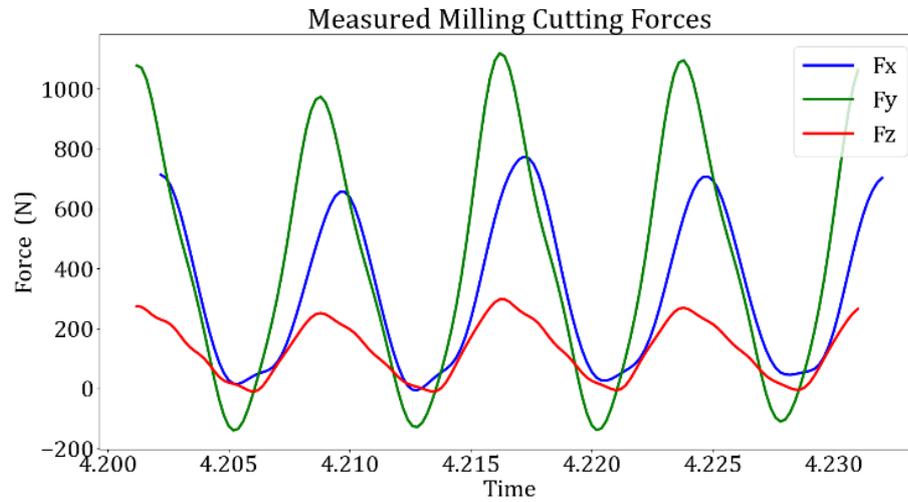


Figure 20. One revolution of measured cutting forces.

Figure 20. illustrates the measured cutting forces with dynamometer. The cutting parameters which have been used for conducting this test are spindle speed is 2000(rpm), Feed is 1600 (mm/min), axial depth is 4(mm), radial depth of cut is 4(mm) and the workpiece material is 1050 steel. This measured data has been used for input of proposed algorithm and the ML algorithm can predict that the tool diameter is 16(mm) with 4 cutting flutes, helix angle is  $35^\circ$  and rake angle is  $7^\circ$ . The predicted tool geometry parameters are the same as the tool which has been used for conducting this test.

Figure 21. shows the other measured cutting forces. In this test spindle speed is 1000(rpm), Feed is 1000(mm/min), axial depth is 7(mm), radial depth of cut is 2(mm) and the workpiece material is 1050 steel. The proposed algorithm predict that tool diameter is 20(mm) with 4 cutting flutes, helix angle is  $30^\circ$  and rake angle is  $5^\circ$ . The predicted tool geometry parameters are the same as the tool which has been used for conducting this test.

15 tests have been conducted to assess the accuracy of the proposed algorithm, which includes four parameters of tool geometry: diameter, tooth number, helix, and rake angle of the tool. The algorithm yielded 56 correct predictions of tool geometry parameters, showcasing an impressive 93% accuracy by utilizing real measured data.

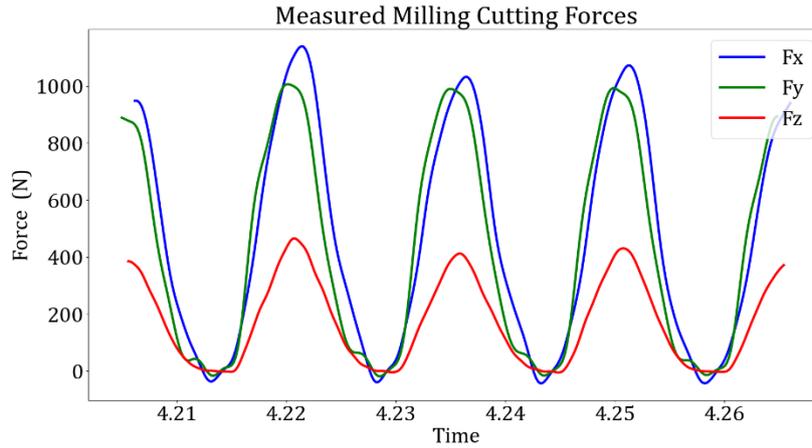


Figure 21. One revolution of measured cutting forces.

#### 4.4. Summary

The main goal of this paper has been to show how using process simulation-based machine learning algorithms can effectively monitor and detect tool-related faults during milling processes. The approach proposed in this study eliminates the need for expensive and time-consuming lab tests by training machine learning models with milling process simulation data.

Throughout this investigation, a range of machine learning algorithms underwent testing using two evaluation metrics. Notably, the random forest algorithm demonstrated superior performance when contrasted with alternative methods for handling such input and output parameters.

Based on the outcomes of this research, the algorithm is capable of achieving a 94 percent accurate prediction rate for tool-related faults. Attaining such a high level of accuracy in predicting tool-related faults solely through the utilization of simulation data can enhance the viability of monitoring systems. This method holds the potential to eliminate the necessity for a substantial volume of tests, a common requirement in typical monitoring applications. These findings have been supported by actual measurement data, with a notable accuracy rate of 93 percent in the predictions.

Although the results are promising, it is essential to continue with further research and improvements. To elevate the overall performance and reliability of the fault monitoring

system, future endeavors could delve into integrating supplementary data sources and adopting advanced machine learning algorithms. Furthermore, the scope of this work can be expanded to encompass process and machine-related faults in forthcoming studies.

This paper has resulted in a notable stride forward in the realm of tool condition monitoring through the application of a process simulation-based algorithm. By streamlining the detection of tool-related faults in a more efficient and cost-effective manner, the demand for extensive laboratory testing has been minimized. This approach holds the capacity to bring about a transformative impact on the domain of tool condition monitoring, thereby enhancing manufacturing processes as a cohesive whole, all the while fostering ongoing research and development efforts.

## 5. MILLING PROCESS MONITORING BASED ON INTELLIGENT REAL-TIME PARAMETER IDENTIFICATION FOR UNMANNED MANUFACTURING

### 5.1. Parameter identification using PBML algorithms

To identify milling process conditions and status during operation, ML algorithms are utilized to extract process parameters from collected cutting forces in real-time. For this purpose, two interrelated steps are employed, as illustrated in Figure 22.

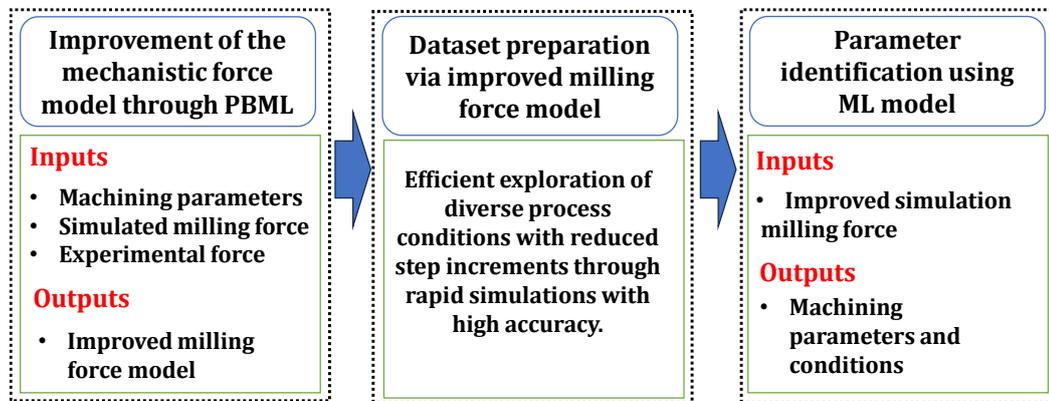


Figure 22. Flowchart of the parameter identification process.

- *Developing a PBML model to improve the accuracy of milling force predictions:* As discussed in the preceding section, the primary challenge in ML training lies in data set preparation. Conducting experiments for this task is both time-consuming and expensive in terms of the computational resources, such as time and memory. Simulation models often lack the accuracy required to generate reliable results for ML training. To overcome this challenge, a hybrid PBML model has been developed to enhance simulation results and effectively address the aforementioned challenge.

For this purpose, the cutting forces were simulated using a linear edge force model [125], and subsequently they were integrated into the PBML model as input features, along with the thermo-mechanical properties of various materials, milling tool geometries and different cutting parameters such as feed rate, axial/radial depth of cuts and measured cutting forces. Integrating mechanistic milling models into the ML framework enables

creating a robust relationship between cutting forces and input features. With this approach, the PBML model provides enhanced simulated milling forces, which are then used in the dataset preparation required for training the ML for parameter identification in the second step. Using simulation alongside ML algorithms eliminates the need for numerous costly and time-consuming experiments.

- *Developing a second-layer ML model for parameter identification:* In this step, the ML model is trained based on the accurate milling force simulations obtained from the previous step. Afterwards, a reverse procedure is employed to predict the cutting parameters based on the real-time cutting forces and ML model.

It should be mentioned that three various ML algorithms were employed for training the models, including support vector machine (SVM) [12], random forest (RF) [113] and least square boosting (LSBoost) [103]. Bayesian optimization [126], a powerful technique in ML and statistical modeling, was employed to select the optimum values for the hyperparameters of ML algorithms, as their accuracy is strongly dependent on these values. By combining probabilistic models with an acquisition function, this algorithm intelligently explores and exploits the parameter space, selecting the most promising configurations iteratively. As Bayesian optimization navigates the search space efficiently, it is particularly useful for optimizing complex and expensive functions, which makes it well suited for hyperparameter tuning. By balancing exploration and exploitation, it can provide a faster convergence to optimal solutions (see Table 10).

Table 10. Hyperparameters of ML models and their optimum values.

<b>ML Models</b>	<b>Hyperparameters</b>	<b>ADOC</b>	<b>RDOC</b>	<b>Feed rate</b>	<b>Tool Radii</b>	<b>No. of teeth</b>
	Minimum leaf size	4	5	3	4	2
<b>RF</b>	No. predictors to sample	21	18	19	17	14
	No. of Trees	24	24	24	24	24
	In bag Fraction	0.8425	0.8142	0.7613	0.8664	0.8124

	No. of Learning Cycles	417	232	385	462	364
	Learning rate	0.2362	0.2176	0.1456	0.2438	0.1859
<b>LSBoost</b>	Minimum leaf size	48	37	31	42	24
	Maximum No. of splits	21351	19824	19215	20457	5487
	No. variables to sample	6	7	4	4	5
	Box Constraint	982.67	943.09	992.83	895.41	886.47
<b>SVM</b>	Epsilon	14.634	13.751	9.231	8.524	7.546
	Kernel Function	Gaussian	Gaussian	Gaussian	Gaussian	Gaussian

## 5.2. Results and discussion

### 5.2.1. Improving accuracy of the milling force model predictions

As previously mentioned, to enhance the accuracy of milling force simulations through ML algorithms, a limited set of measurements were taken. To achieve this, a comprehensive array of cutting parameters, predominantly employed in industrial applications, were selected for conducting milling force measurements. To conduct a suitable set of experiments, the composite central design (CCD) method was utilized. Two different end mills were used for these tests - with 20mm and 16mm diameters and cutting teeth of 4 and 3, respectively. The cutting parameters of these measurements are detailed in Table 11.

Table 11. Training cutting parameters for improved milling force prediction with ML algorithm.

<b>Cutting parameters</b>	<b>Levels</b>
<b>Spindle speed (rpm)</b>	2500, 3500, 4500
<b>Feed rate (mm/rev.tooth)</b>	0.025, 0.05, 0.1, 0.15
<b>Axial depth of cut (mm)</b>	1, 3, 6, 10, 15
<b>Radial depth of cut (mm)</b>	1, 2, 4, 8, 10
<b>Material</b>	AL7075-T6

Cutting forces used for ML training were measured using a Piezo-electric Dynamometer 9257BA, an amplifier, and the NI USB-6259 data acquisition system.

Following the collection of measured cutting force data and the training of ML models, the established PBML model was employed to correlate experimentally measured cutting forces with simulated milling forces using 2.40 GHz Intel Core i7-13700H CPU and 32GB RAM. The model was trained by LSBoost, SVM and RF algorithms, and was subsequently employed to predict the cutting forces of an unseen dataset. It should be mentioned that 70% of the dataset was used for training, while 15% was allocated for validation and another 15% for testing. Comparison of the obtained results with experimentally measured forces reveals that the proposed PBML model remarkably improves the accuracy of the mechanistic models, where the correlation coefficient ( $R^2$  value) is more than 97% for the unseen datasets. The LSBoost model yielded the lowest root mean square error (RMSE), followed by RF and SVM, respectively. An example of the obtained results (i.e., enhanced cutting forces) is demonstrated in Figure 23.

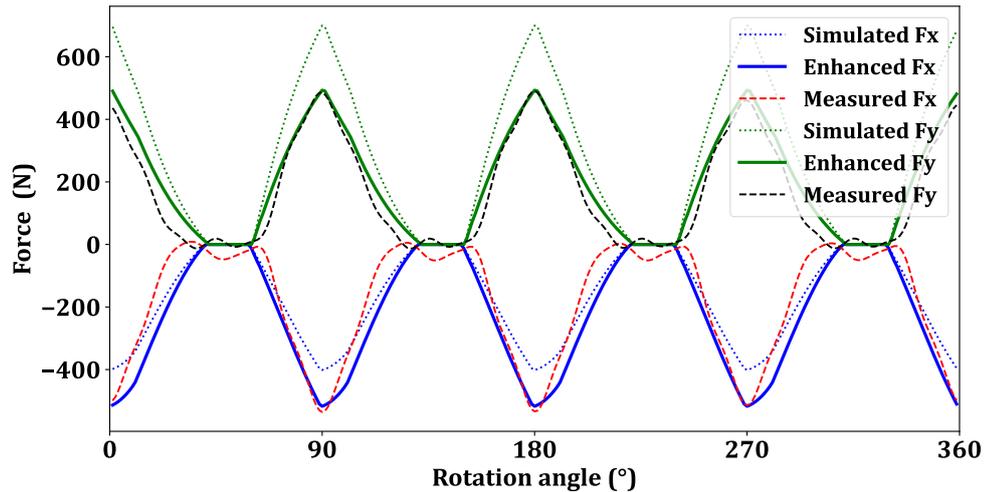


Figure 23. Comparison of the measured cutting forces with simulations and enhanced simulations using PBML. (tool diameter is 16mm, number of teeth is 4, runout is 10  $\mu\text{m}$ , axial and radial depth of cuts are 4mm, feed rate is 0.2 mm/rev.tooth, and spindle speed is 2000rpm)

### 5.2.2. Parameter identification

After the accuracy of the mechanistic model was improved, it was utilized to generate a training dataset, a crucial step in the subsequent parameter identification process. The low calculation time of these simulations allows the exploration of a wide range of conditions within a short period, a task which is impractical with experimental data only due to cost and time constraints. This expanded dataset played a pivotal role in refining the model's ability to identify parameters effectively. The output of this model was the estimated milling conditions, including feed rate, axial/radial depth of cut, tool diameter, number of teeth etc. It should be mentioned that the spindle speed was identified by applying Fast Fourier Transform (FFT) on cutting force signals and using the identified number of teeth by ML.

As illustrated in Table 12, the proposed model has yielded exceptionally accurate results, with the predicted parameters closely aligning with the actual values. The predictions accuracies exceed 95% for all machining parameters using the developed PBML model, and it demonstrates superior performance in parameter identification. The obtained results demonstrate the effectiveness of PBML in achieving a high level of accuracy in the context of parameter identification.

Table 12. Parameter identification performance of ML algorithms.

ML Models	Metrics	Axial depth of cut	Radial depth of cut	Feed rate	Tool radius	No. of Teeth
RF	R <sup>2</sup> (%)	95.03	97.88	96.14	94.20	97
	RMSE	0.036	0.21	0.024	0.015	0.01
LSBoost	R <sup>2</sup> (%)	97.89	98.65	97.32	98.47	98
	RMSE	0.032	0.14	0.015	0.01	0.008
SVM	R <sup>2</sup> (%)	95.11	96.63	95.09	95.74	97
	RMSE	0.041	0.23	0.027	0.014	0.01

To clarify the precision of the obtained outcome, a statistical error analysis was conducted to portray the distribution of disparities between the actual and predicted values of machining parameters. The results, illustrated in Figure 24. The statistical error analysis of the identification of a) axial depth of cut (ADOC) and b) radial depth of cut (RDOC). The percentage error between the milling force predictions and the measured values were determined for over 50 different unseen data. Figure 24, reveal that the error distribution is tightly constrained within a narrow band, with the majority of deviations falling between -5% and 5%. This interval encompasses 90% of the observations. The model's adeptness in capturing and minimizing errors within this specified range not only reinforces the overall robustness of the parameter identification process but also underscores the developed ML approach as a valuable tool for achieving a high level of accuracy in intricate machining scenarios.

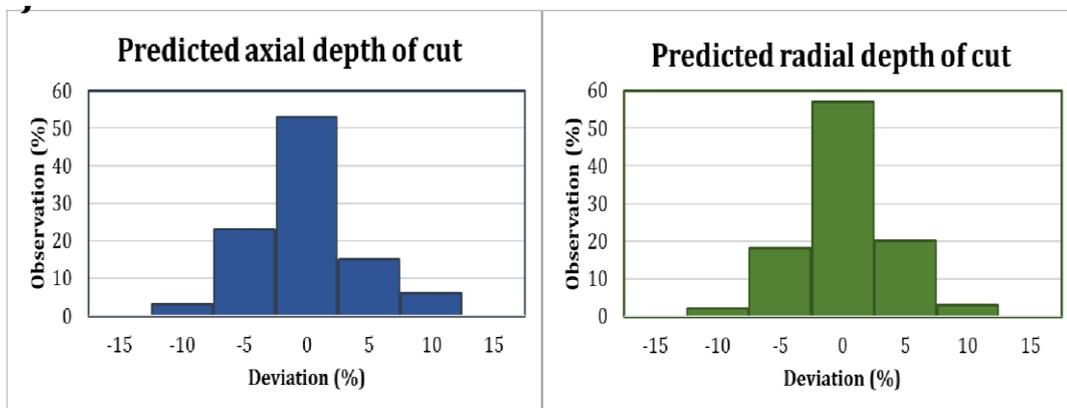


Figure 24. The statistical error analysis of the identification of a) axial depth of cut (ADOC) and b) radial depth of cut (RDOC). The percentage error between the milling force predictions and the measured values were determined for over 50 different unseen data.

### **5.2.3. Experimental verification**

To experimentally verify the proposed method, the accuracy of the algorithm was evaluated on milling of a complex part geometry. For this purpose, the test workpiece illustrated in Figure 25 was prepared to include continuous variations in both axial and radial depth of cuts on the AL7075-T6 block. Furthermore, the feed rate was altered in three steps over the cutting length. This scenario is referred to as case 1.

To identify varying process parameters, the real-time measured cutting forces by dynamometer were fed to the trained ML model (as the input features) in each 0.3 second. The sampling rate of the system is determined based on the computation time of the ML model. Subsequently the model predicted the cutting parameters in different cutter locations. Comparison of actual and identified process parameters are shown in Figure 26 and Figure 27 (Case 1). Comparison of the actual cutting parameters with predicted values (through real-time cutting forces) reveals a high level of precision, where the estimation accuracy for axial depth of cut is 97.9%, for radial depth of cut is 98.7%, and for feed rate is 97.3%. The highest RMS error is 0.14, illustrating the effectiveness of the proposed methodology in capturing the intricate variations in cutting conditions throughout different regions of the specimens (see Figure 26 and Figure 27, case 1).

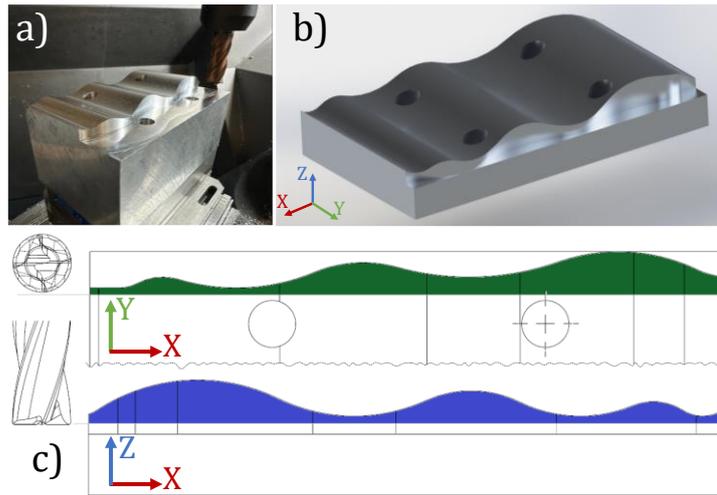


Figure 25. a) Experimental setup for verification b)3D view and c) top and side views of the test workpiece.

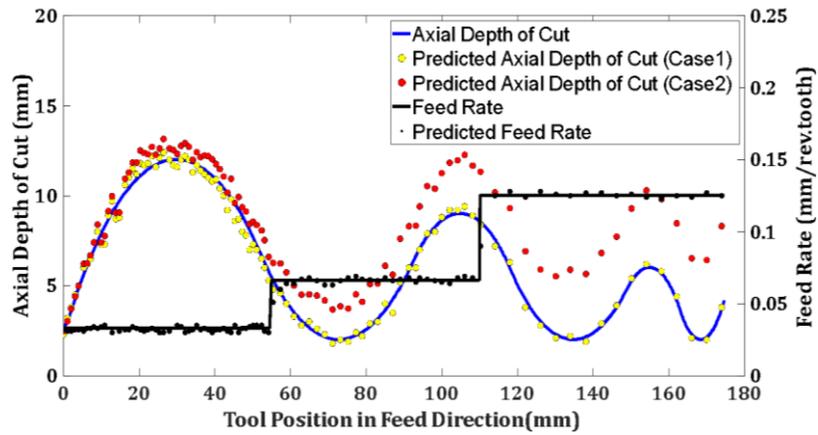


Figure 26. Comparison of identified & actual axial depth of cut and feed rate.

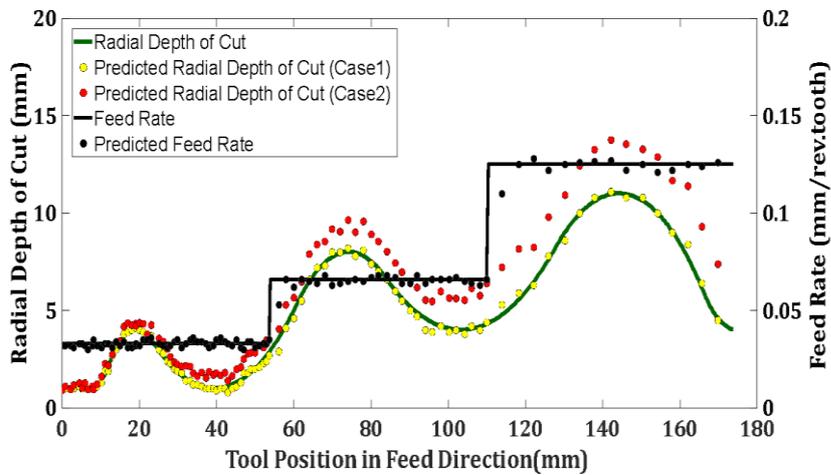


Figure 27. Comparison of identified and actual radial depth of cut and feed rate.

The remarkable precision observed across various parameters confirms the proficiency of the model proposed in thoroughly comprehending and predicting complex patterns in the cutting process. Additionally, the results obtained suggest that the proposed approach functions as a reliable and adaptable tool for accurately identifying parameters in practical machining applications, utilizing real-time cutting forces. This can also be considered as digitalization of an existing machining process to obtain the required conditions for further analysis and simulations.

### **5.3. Production Applications**

#### **5.3.1. Monitoring and fault detection**

As mentioned earlier, the main advantage of the proposed method, in contrast to previous monitoring systems, is its capability to identify accurate cutting parameters for detection of various faults and their sources. These faults may originate from errors in workpiece clamping, tool definition offsets, wrong fixture position and/or offsets, errors in workpiece and machine tool coordinate transformation etc. The real-time detection of such discrepancies is crucial for machining operations and investigated in a case study. The example case involved clamping of the workpiece with angular displacements around both  $X$  and  $Z$  axes, resulting in discrepant axial and radial depth of cut values (i.e., different from the desired values), representing a possible fixturing error, which is referred to as case 2. The cutting forces, as measured by the dynamometer, were then utilized for parameter identification using the proposed PBML model outlined in preceding sections. The results, as depicted in Figure 26 and Figure 27(case 2), demonstrate that the model has well captured the discrepancy of  $1.5^\circ$  and  $1^\circ$  in axial and radial directions, respectively, highlighting the efficiency of the developed approach in accurately detecting workpiece-clamping faults.

#### **5.3.2. Parameter optimization**

As an important application of this method, the identified parameters can be utilized for real-time cutting parameter optimizations. For instance, the feed rate significantly influences machining time, cutting forces, and overall production efficiency. Feed rate

optimization not only minimizes process time but also serves as a means to maintain cutting forces at a desired level respecting machine, tool and part constraints eliminating overloads while providing uniform quality, as well. Utilizing the exponential cutting force model, Equation (12) [127] was employed to determine the optimal feed rate from measured cutting forces.

$$f_o = f \left( \frac{F_{tar}}{F_E} \right)^{\frac{1}{1-p}} \quad (12)$$

where,  $F_E$  is the estimated cutting force by PBML,  $F_{tar}$  is the target force,  $f$  and  $f_o$  are estimated and optimal feed rate, and ‘ $p$ ’ is a material-dependent constant found by exponential cutting force model.

The obtained result for feed rate optimization is illustrated in Figure 28. In this context, the optimization objective was to ensure that the resulting cutting force remains below the specified target force value of 600 N. In addition to the constant resultant cutting force, the optimized feed rate values lead to a 27% reduction in the cutting time.

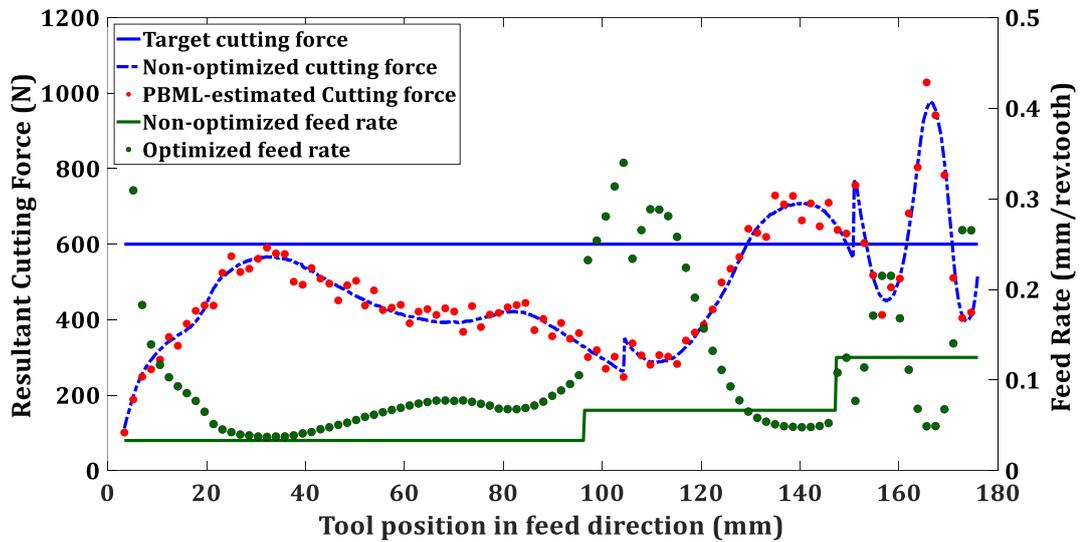


Figure 28. Feed rate optimization utilizing PBML-estimated resultant cutting forces.

### 5.3.3. Current data from CNC controller

The developed approach is versatile, supporting various sensors for implementation in different situations. Using dynamometers can be challenging due to the obstacles they introduce, the cables can create operational issues, moreover they are expensive and

vulnerable to damage in the production environment. Therefore, using alternative sensors for real-time identification of cutting parameters is more practical. The direct utilization of servo current data from the CNC controller is a convenient approach, though it suffers from a major drawback of insufficient sampling rates, particularly at higher spindle speeds. Consequently, the ongoing project is focused on exploring the use of CNC controller data for parameter identification at high spindle speeds. In this context, the servo current data captured is employed to derive cutting forces through a ML model. Figure 29 illustrates a comparison between the cutting forces derived from the current data and those acquired by the dynamometer where a good agreement is observed. Subsequently, the derived cutting forces can be utilized for identifying the process parameters.

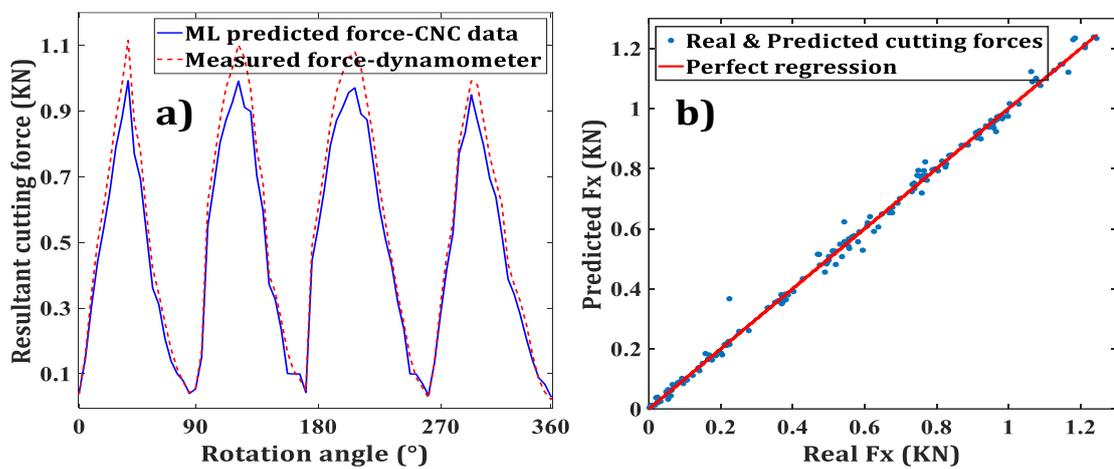


Figure 29. a) Comparison of cutting forces collected by dynamometer and ML-predicted forces using CNC controller data, b) Regression curve for unseen test data (The tooth passing frequency is equal to 66.66 Hz).

#### 5.4. Summary

This study aims for the development of an intelligent monitoring system through real-time fault detection using a novel ML system developed based on a hybrid approach.

- Highly accurate simulation results obtained by PBML, based on the linear edge force model, are used instead of experimental data to train the ML model.

- The proposed approach demonstrates remarkable accuracy in predicting the machining parameters, exceeding 96% in real-time. The statistical error analysis indicates a notably confined error distribution.
- The high prediction accuracy of the proposed method in real-time has been verified by the machining of a complex free-form workpiece.
- The proposed approach finds applicability across various unmanned manufacturing applications serving purposes such as process monitoring, fault detection and parameter optimization, as exemplified in the case studies.
- The suggested method exhibits notable adaptability for deployment in industrial applications, such as utilizing CNC controller current signals instead of a dynamometer.

## 6. PHYSICS-INFORMED TOOL WEAR PREDICTION IN TURNING PROCESS: A THERMO-MECHANICAL WEAR-INCLUDED FORCE MODEL INTEGRATED WITH MACHINE LEARNING

### 6.1. Methodology

This section presents an overview of the mechanistic modeling approach used to estimate cutting forces, considering both unworn and worn cutting tool conditions. First, thermo-mechanical modeling of the primary, secondary, and third deformation zones is introduced, neglecting the influence of flank wear but incorporating the effect of edge forces. Subsequently, the proposed model for incorporating the effect of tool wear on cutting forces is presented, and the application of this model for the turning process is discussed, considering the effect of the nose radius. Finally, the details of the physics-informed ML model used to improve the accuracy of the mechanistic model are presented.

#### 6.1.1. Modeling of the primary, secondary and third deformation zones

In this section, the thermo-mechanical modeling of cutting forces in the primary, secondary, and third deformation zones is introduced. To facilitate the mathematical tractability of the model, the cutting tool was partitioned into six regions, as illustrated in Figure 30a. Regions 1-3, located on the rake face, are responsible for chip formation; region 4, being on the hone face, is responsible for ploughing; and the regions 5-6 represent the flank contact due to elastic recovery.

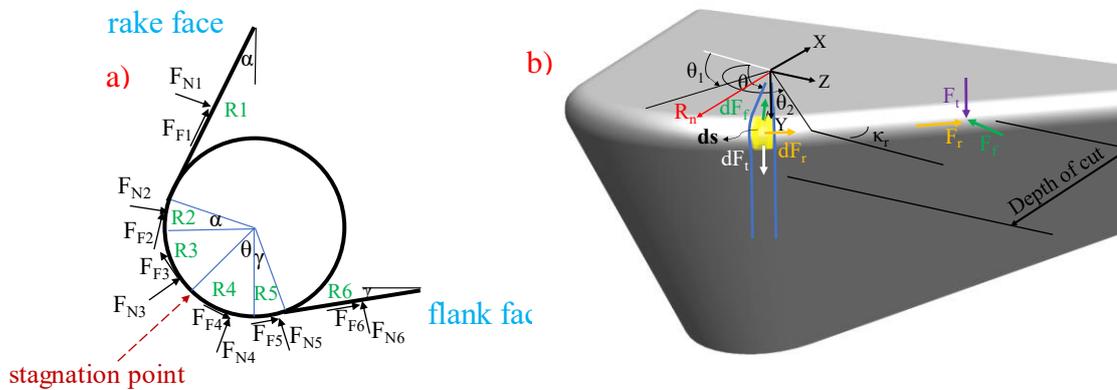


Figure 30. a) Representation of cutting-edge including rake, hone and flank faces,  
b) 3D tool model considering nose radius effect.

Unlike conventional force models, the model used in this study does not rely on constant cutting coefficients to calculate cutting forces. Instead, it employs local coefficients derived from the local pressure and shear stress distributions on the rake, hone, and flank faces. To calculate the stress distribution on these faces, the shear stress at the exit of the shear zone ( $\tau_l$ ) has to be first determined. By considering the inertia effects, from the equations of motion for continuous chip condition,  $\tau_l$  is calculated by Equation (13):

$$\tau_1 = \rho (V_c \cos\lambda_s \sin\phi_n)^2 \gamma_1 + \tau_0 \quad (13)$$

where,  $\rho$  is the material density,  $V_c$  is the cutting speed,  $\lambda_s$  is the inclination angle,  $\phi_n$  is the normal shear angle, and  $\gamma_l$  is the shear strain at the exit of the shear band.  $\tau_0$  is the shear stress at the entry of the primary deformation zone, which is calculated using the Johnson-Cook model, with strain rate and temperature boundary conditions applied at the entry and exit of the shear band [128]:

$$\left\{ \begin{array}{l} \dot{\gamma} = \dot{\gamma}(\gamma, \tau_0) = \dot{\gamma}_0 \exp\left(\frac{\tau\sqrt{3}}{m g_1(\gamma)g_2(T)} - \frac{1}{m}\right) \\ g_1(\gamma) = A + B \left(\frac{\gamma}{\sqrt{3}}\right)^n \\ g_2(T) = 1 - \left(\frac{1 - T_{ref}}{T_m - T_{ref}}\right)^v \\ T = T_w + \frac{\beta}{\rho c} [\rho (V \cos\lambda_s \sin\phi_n)^2 \frac{\gamma^2}{2} + \tau_0 \gamma] \end{array} \right. \quad (14)$$

where,

$$\frac{d\gamma}{dz} = \frac{\dot{\gamma}(\gamma, \tau_0)}{V \cos\lambda_s \sin\phi_n} \quad (15)$$

$$\gamma_h = \frac{\cos\alpha_n}{\sin\phi_n \cos\eta_s \cos(\phi_n - \alpha_n)} \quad (16)$$

$$\int_0^{\gamma_h} \frac{V \cos\lambda_s \sin\phi_n}{\dot{\gamma}(\gamma, \tau_0)} d\gamma = h \quad (17)$$

where,  $\gamma$  is shear strain,  $T_r$  is reference temperature,  $T_m$  is the melting temperature,  $n$  is the strain hardening exponent,  $m$  is the strain rate sensitivity,  $v$  is thermal softening coefficient,  $A$  and  $B$  are material constants,  $\eta_s$  is the shear flow angle, and  $h$  is the thickness of the primary shear band.

After estimating the shear stress at the exit of the primary shear zone, the pressure distributions on the rake, hone and flank faces are calculated. For the rake face, most of the researchers have employed a decreasing quadratic pressure distribution, as represented by equation (18). While, for the hone and flank faces, there is no universally accepted solution, and different scenarios are tested by researchers [120]. Therefore, in this study various distribution functions were examined, including linear, quadratic and cubic for both “decreasing” and “increasing-decreasing” patterns, as depicted in Figure 31.

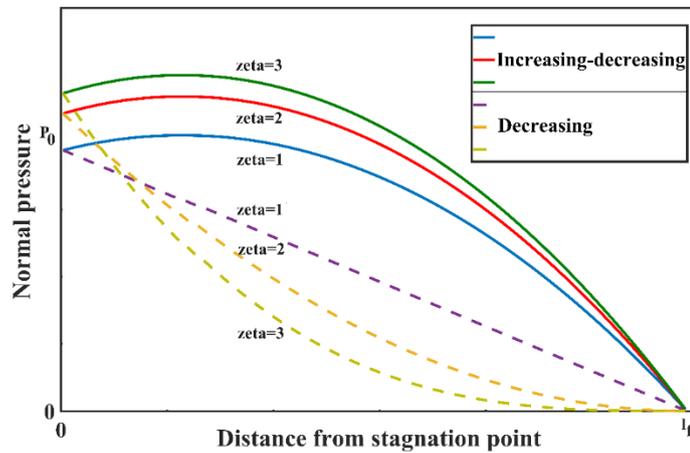


Figure 31. Different pressure distribution patterns on hone and flank faces.

$$P(x) = P_0 \left(1 - \frac{x}{l_f}\right)^\zeta \quad (18)$$

$$P_0 = \tau_1 \frac{h_1(1 + \zeta)}{l_c} \frac{\cos\eta_s \cos\beta_n}{\sin\phi_n \cos\eta_c \cos(\phi_n + \beta_n - \alpha_n)}$$

$$P(x) = a\chi^2 + b\chi + c$$

$$a = \frac{P_0}{2l_f l_1 - l_f^2}; \quad b = -2al_1; \quad c = P_0; \quad l_1 = R_h \cdot \theta \quad (19)$$

where,  $l_f$  denotes the contact length in the flank face,  $l_c$  is the contact length in the rake face,  $\zeta$  is the distribution exponent,  $P_0$  is the normal pressure at the stagnation point,  $\beta_n$  is the normal friction angle,  $R_h$  is the hone radius,  $\theta$  is the stagnation angle, and  $\chi$  represents the distance from stagnation point. The stagnation point separates the secondary deformation zone from the third deformation, as illustrated in Figure 30a.

The equations (20) and (21) describe the normal and frictional forces acting on the cutting tool across the six previously mentioned regions, i.e. the rake face, hone region, and flank face [120].

$$F_{Ni} = \int_{l_i}^{l_i+l_{i+1}} w \cdot P(\chi) \cdot o(\chi) d\chi \quad (20)$$

$$F_{Fi} = \begin{cases} \int_{l_i}^{l_i+l_{i+1}} \tau_1 \cdot w \cdot o(\chi) d\chi, & \text{sticking region} \\ \int_{l_i}^{l_i+l_{i+1}} \mu \cdot P(\chi) \cdot w \cdot o(\chi) d\chi, & \text{sliding region} \end{cases} \quad (21)$$

$$\begin{cases} \text{if } \mu \cdot P(\chi) > \tau_1; & \text{sticking condition} \\ \text{if } \mu \cdot P(\chi) < \tau_1; & \text{sliding condition} \end{cases} \quad (22)$$

where,  $F_N$  is the normal force,  $F_F$  is the friction force,  $w$  is the depth of cut and  $o(x)$  is the orientation function for the partitioned regions. The shearing forces acting on the rake face ( $F_{tc}$  and  $F_{fc}$ ) are calculated by equation (23); and the edge forces acting on hone and flank faces ( $F_{te}$  and  $F_{fe}$ ) are calculated by equation (24), respectively:

$$F_{tc} = \sum_{i=1}^3 F_{Niz} + \sum_{i=1}^3 F_{Fiz} \quad (23)$$

$$F_{fc} = \sum_{i=1}^3 F_{Niy} + \sum_{i=1}^3 F_{Fiy}$$

$$F_{te} = \sum_{i=4}^6 F_{Niz} + \sum_{i=4}^6 F_{Fiz} \quad (24)$$

$$F_{fe} = \sum_{i=4}^6 F_{Niy} + \sum_{i=4}^6 F_{Fiy}$$

where, the subscript *c* and *e* refer to shearing forces and edge forces, respectively. The sum of these forces represents the total cutting forces.

### 6.1.2. Modeling flank wear effect on cutting forces

To investigate the wear effect on cutting forces, a virtual wear is simulated, and the distribution of forces acting on the worn area is recalculated. The formation of new contact regions due to wear alters stress distributions, friction, and normal forces, necessitating recalculation and updates to accurately capture these changes. A linear wear pattern parallel to the cutting speed, which is commonly used by other researchers [89,129], was employed in this study. The wear effect on the cutting forces can be addressed by recalculating and updating the forces acting on the worn zone, as illustrated in Figure 32 and equation (25):

$$F_{N4} = \int_{l_{st}}^{l_{end}} w \cdot P(x) dx \quad (25)$$

$$F_{F4} = - \int_{l_{st}}^{l_{end}} \tau(x) \cdot w dx$$

where,  $l_{st}$  and  $l_{end}$  are the start and end points of the flank wear.  $\tau$  is the shear stress, which is dependent on the friction type (i.e., sliding or sticking). As previously explained in equation (22), at points where  $\mu P(x)$  is greater than  $\tau_I$ , the sticking condition exists and  $\tau$  is equal to  $\tau_I$ ; while for the rest of the contact length, sliding condition exists and

consequently  $\tau$  is equal to  $\mu P(x)$ . An important consideration is that the friction coefficient of the worn surface inherently differs from that of the unworn surface. Hence, the friction behavior on the worn surface was investigated using an iterative numerical approach. For this purpose, the friction coefficient was calibrated to minimize the relative error between the experimental and analytical results. This involved performing wear modeling with various friction coefficient values on the worn surface. The value that yielded the least difference between measured and model-estimated forces was then selected as the friction coefficient for the worn surface. Also, the investigation included examining different pressure distributions on the hone and flank faces, as outlined in the previous section.

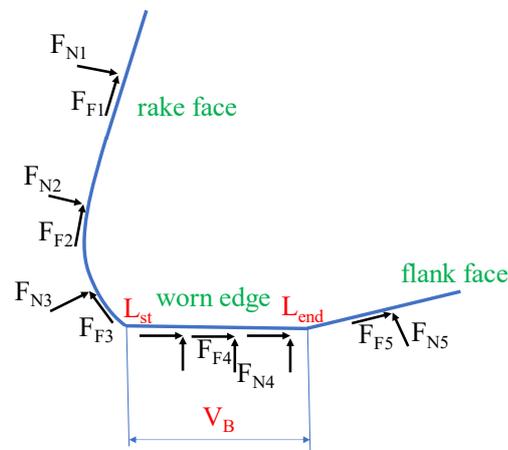


Figure 32. Schematic representation of flank wear.

### 6.1.3. Modeling of turning process

One of the applications of the proposed model is for the turning operation, where the effect of nose radius and cutting-edge angle also needs to be considered. For the sake of the mathematical simplicity, the cutting edge was partitioned into two regions: R1, representing the straight edge, and R2, representing the nose region, as depicted in Figure 1b. In the first region, the direction and magnitude of the forces along the straight edge don't change, unlike the second region in which the local edge angles are varying [130]. Therefore, the nose region is subdivided into infinitesimal elements with a length of  $ds$  ( $=R_n d\theta$ ) and a local edge angle of  $\kappa_r(\theta)$ . The presence of the nose radius results in local variations in cutting angles [131]. Even if the tool has no global inclination angle, the global rake angle leads to local inclination and rake angles along the nose radius, as

explained by equation (26).

$$\begin{aligned}\alpha_n^j &= 2\sin^{-1}\left(\cos(\kappa_j - \kappa_r) \sin\left(\frac{\alpha_n}{2}\right)\right) + 2\sin^{-1}\left(\sin(\kappa_j - \kappa_r) \sin\left(\frac{i}{2}\right)\right) \\ i^j &= 2\sin^{-1}\left(\sin(\kappa_j - \kappa_r) \sin\left(\frac{\alpha_n}{2}\right)\right) - 2\sin^{-1}\left(\cos(\kappa_j - \kappa_r) \sin\left(\frac{i}{2}\right)\right)\end{aligned}\quad (26)$$

where,  $\alpha_n^j$  represents the local normal rake angle, and  $i^j$  is the local inclination angle for the  $j^{\text{th}}$  element;  $\alpha_n$  and  $i$  are global normal rake and inclination angles, respectively.

The feed and radial forces are calculated using a similar approach as explained in the previous sections. These elemental forces are then transformed into the global X, Y, and Z coordinate system to obtain the total cutting forces in the global (dynamometer) reference system, as described by equations (27)-(29).

- **For region 1:**

$$F_x^{R1} = \frac{1}{w} [-F_f \cos \kappa_r + F_r \sin \kappa_r] \left( \frac{w - R_n(1 - \cos \kappa_r)}{\sin \kappa_r} \right) \quad (27)$$

$$F_z^{R1} = \frac{1}{w} [F_f \sin \kappa_r + F_r \cos \kappa_r] \left( \frac{w - R_n(1 - \cos \kappa_r)}{\sin \kappa_r} \right)$$

- **For region 2:**

$$dF_x^{R2} = -dF_f^{R2} \sin \theta + dF_r^{R2} \cos \theta \quad (28)$$

$$dF_z^{R2} = dF_f^{R2} \cos \theta + dF_r^{R2} \sin \theta$$

$$F_q^{R2} = \sum_{j=1}^N dF_{q,j}^{R2}, \quad q = x, z \quad (29)$$

where,  $N$  is the number of elements in the nose region, and  $F_x$ ,  $F_y$  and  $F_z$  represent global radial, tangential and feed forces, respectively.

The total cutting forces on the tool edge are calculated by summing the forces acting on the regions 1 and 2:

$$F_q^{total} = F_q^{R1} + F_q^{R2}, \quad q = x, y, z \quad (30)$$

As a sample, the cutting forces estimated by the wear-included turning model is illustrated in Figure 33 for a specific cutting condition and wear length value (e.g., feed rate: 0.05

mm/rev; depth of cut: 0.5 mm; cutting speed: 100 m/min; nose radius: 0.2 mm; hone radius: 15  $\mu\text{m}$ , wear length= 190  $\mu\text{m}$ ). Complete results for other cutting conditions and wear lengths are presented in the Results section.

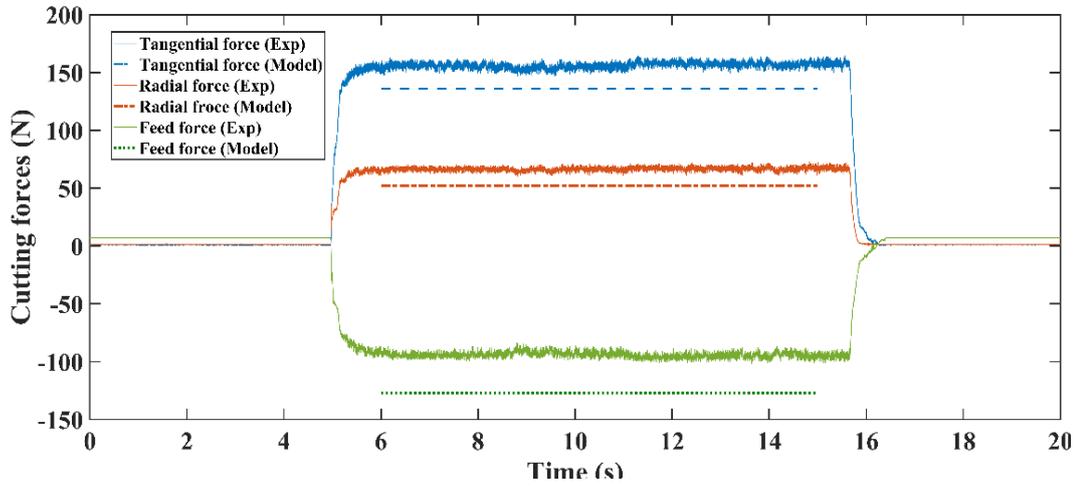


Figure 33. A sample of measured and estimated cutting forces (feed rate: 0.05 mm/rev; depth of cut: 0.5 mm; cutting speed: 100 m/min; nose radius: 0.2 mm; hone radius: 15  $\mu\text{m}$ , wear length= 190  $\mu\text{m}$ ).

#### 6.1.4. Physics-informed machine learning model

Conventional machine learning models, relying solely on experimental data, suffer from the high costs and extensive time required for data collection. This limitation has restricted the application of ML to academic research rather than real production environments, primarily due to the challenges in data preparation. To bridge the gap between industry and research, a novel physics-informed machine learning (PIML) [50] model was developed to predict cutting forces in the turning process. This hybrid model is not solely trained on experimental data but also incorporates estimations derived from mechanistic models. In other words, the forces estimated by the mechanistic model serve as additional inputs to the ML model, along with machining parameters and tool geometrical properties. This innovative approach enriches the predictive capabilities of the ML model beyond what could be achieved with experimental data alone. A schematic

representation of the proposed model is illustrated in Figure 34a. As depicted, the inputs of the ML model are feed rate, cutting speed, depth of cut, hone radius, nose radius, wear length and the cutting forces estimated by the mechanistic model. Meanwhile, the experimentally measured cutting forces are the targets of the hybrid model. Such a hybrid physics-informed model, which incorporates the mechanistic models in the training algorithm, not only improves the accuracy of the mechanistic model, but also eliminates the need for performing numerous experimental tests, an essential requirement often associated with data-driven ML models.

Upon verifying the accuracy of the hybrid model, it can be employed in a reverse manner to predict wear length based on cutting forces and machining parameters. Here, the improved wear-included turning model from the previous step is used to generate a reliable training dataset for wear prediction, avoiding numerous expensive and time-consuming experimental wear tests. As demonstrated in Figure 5b, for this reverse ML model, the wear length is considered as the output, while the inputs are the improved cutting forces, machining parameters, and tool geometry.

To achieve the most accurate predictions possible, different ML algorithms were utilized for data analysis, including random forest (RF), support vector regression (SVR), and least square boosting (LSBoost). In this respect, initially hyperparameter optimization was conducted using Bayesian optimization. This algorithm combines probabilistic models with an acquisition function to navigate the parameter space, iteratively selecting the most beneficial configurations. As Bayesian optimization explores the search space effectively, it is specifically well-suited for optimizing complicated and expensive functions, making it appropriate for hyperparameter tuning. By balancing exploration and exploitation, it facilitates faster convergence to optimal solutions [115].

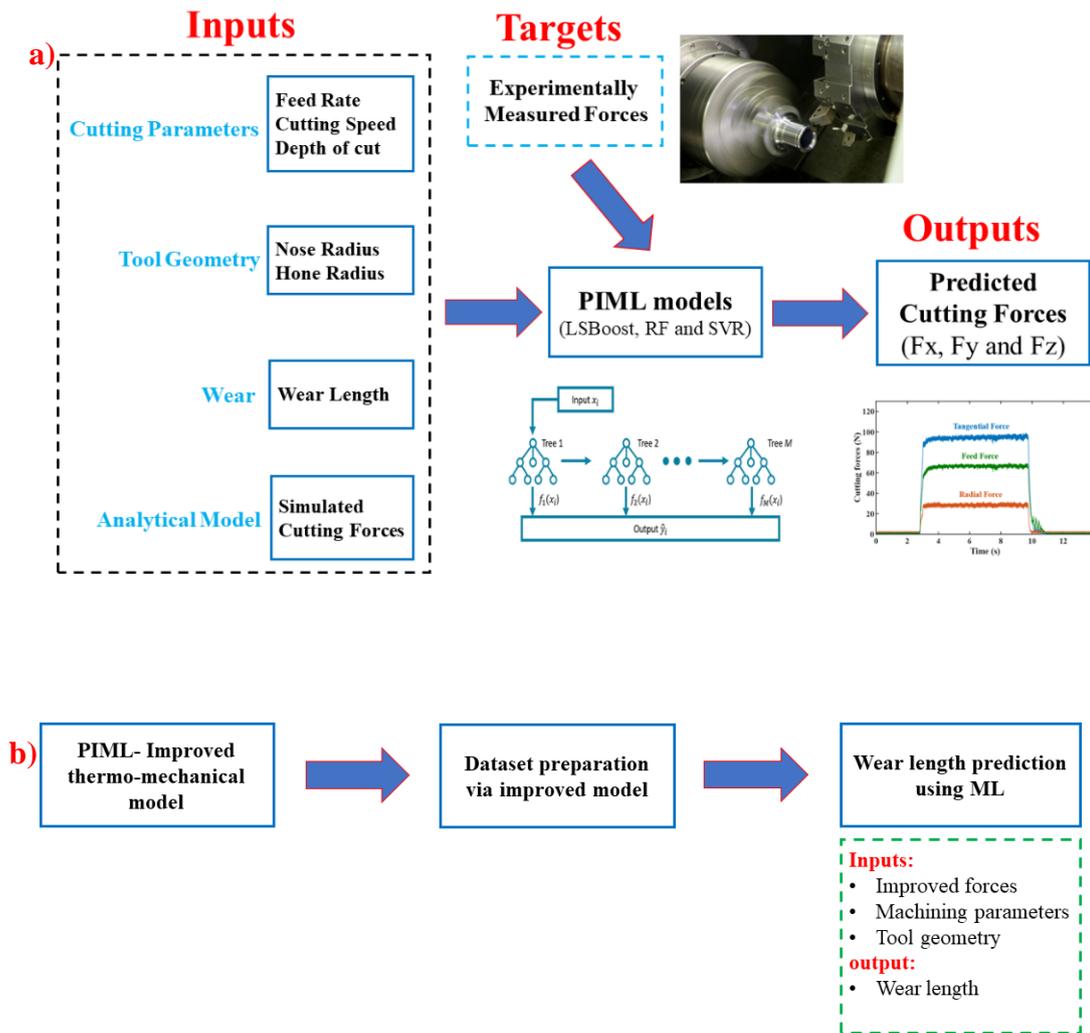


Figure 34. Schematic diagram of the hybrid physics-informed ML model:

- a) Force prediction in the presence of tool wear,
- b) Wear length prediction via reverse ML model.

## 6.2. Experimental set-up

Figure 35 depicts the wear and turning process test setup, where cutting forces were measured using a Piezo-electric Dynamometer 9257BA, an amplifier, and the NI USB-6259 data acquisition system. The workpiece material is Steel 1050, and the cutting tool

is uncoated, brand-new turning inserts from the TPGN 160-883 series with various nose radii of 0.2, 0.4, and 0.8 mm. The wear measurements were performed with Dino-Lite digital microscope and Nano focus  $\mu$ -surf explorer, as shown in Figure 35b.

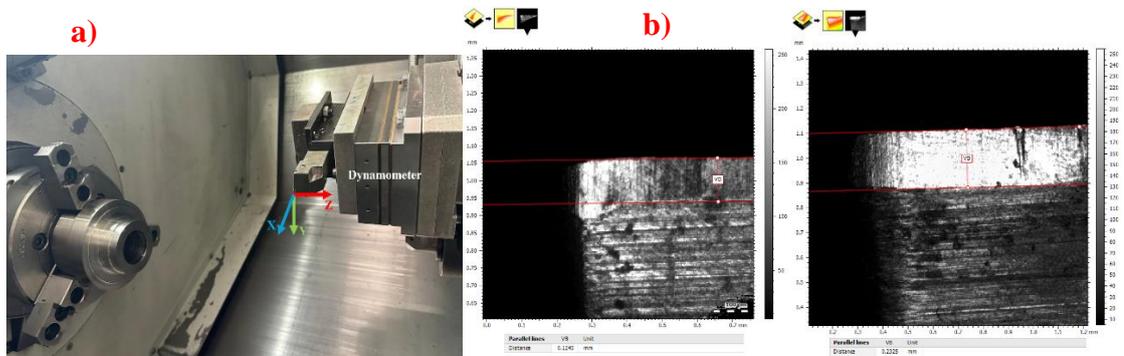
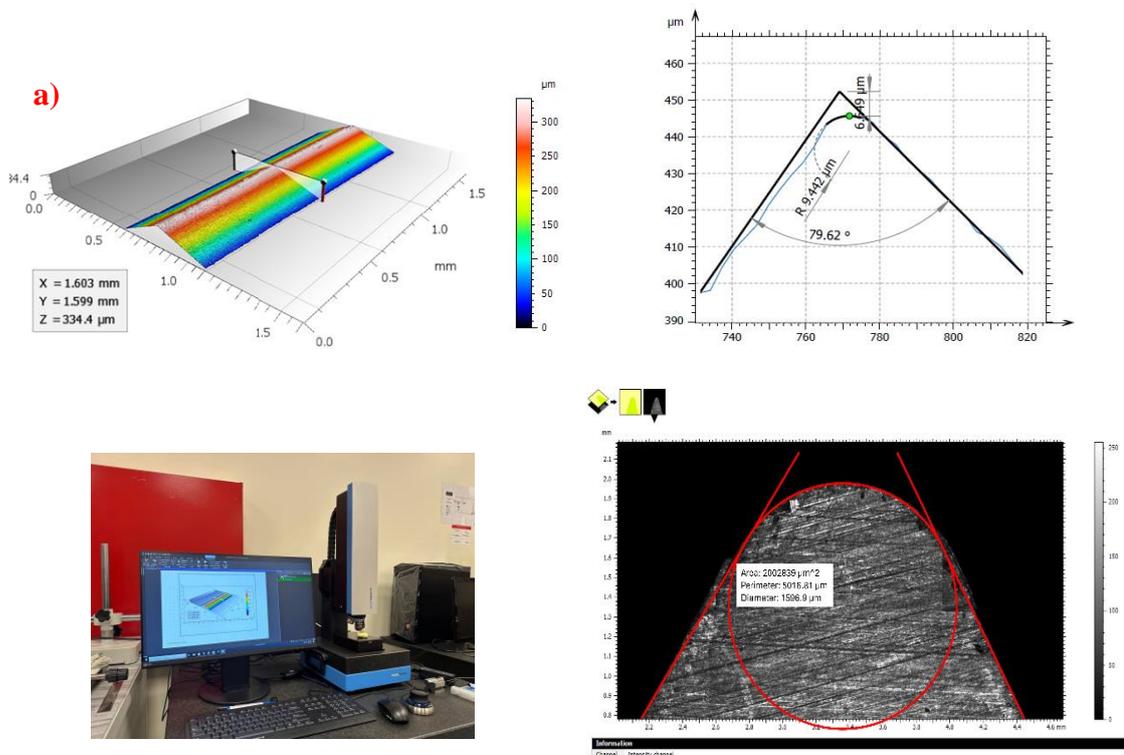


Figure 35. Experimental set-up, a) Lathe machine, MORI SEIKI, NL1500  
b) Tool wear measurement by Nano-focus device.

The hone and nose radius measurements were carried out by Nano-focus measurement machine, as illustrated in Figure 36.



b)

Figure 36. a) Hone radius measurement (10  $\mu\text{m}$ ), b) Nose radius measurement by Nano-focus  $\mu$ -surf explorer

The Composite Central Design (CCD) method was employed to conduct the experiments. This method offers significant benefits, primarily by reducing the total number of the required experimental tests. CCD allows researchers to efficiently explore the effects of various factors with minimal resource expenditure. This efficiency is achieved through the strategic placement of center points and axial points in the design space, which enhances the ability to predict responses with fewer trials. This approach not only conserves resources but also accelerates the experimental process, making it an effective strategy for robust and systematic investigation.

### **6.3. Results and discussion**

#### **6.3.1. Improvement of the turning model accuracy through ML**

The accuracy of the turning model was evaluated without considering flank wear, while incorporating the effects of hone radius and nose radius into the assessment. For this purpose, the experimental tests were conducted using brand-new cutting inserts with varying nose and hone radii, and the measured forces were compared with the model-estimated forces. A sample of the measured cutting forces is depicted in Figure 37, and the comparison of the model-estimated forces and the experimental results is summarized in Table 13. It should be mentioned that for the hone and flank faces, different pressure distribution functions were investigated, considering constant pressure, increasing-decreasing and decreasing, each with different zeta values. Among various functions, two patterns demonstrated better estimations: the increasing-decreasing and decreasing patterns, both with zeta values set to one. As summarized in Table 1, the coefficient of determination ( $R^2$ ) for the analytical model ranges from 67% to 85%, indicating a moderate level of agreement between the model estimations and experimental results. However, this suggests that the model's estimations are not precise enough to reliably predict cutting forces in the turning process. Therefore, as previously mentioned in Section 6.1.4, a hybrid physics-informed machine learning model was developed to

improve the accuracy of the mechanistic model. In the proposed ML approach, the predicted forces by the mechanistic model were also included as the inputs of the ML algorithm, along with the machining parameters and tool geometrical properties. To achieve the most accurate predictions possible, different ML algorithms were utilized for regression analysis, including random forest, support vector regression, and LSBoost. The dataset was first partitioned into three subsets: training, validation, and test data. In this respect, 75% of the dataset was allocated for training purposes, 15% for validation to evaluate model performance during training, and the remaining 15% was kept as unseen test data to assess the generalization performance of the trained model. The dataset was standardized to have zero mean and unit variance. To achieve this, the necessary statistics were computed using the training data, and these statistics were then applied to both the training and test datasets to avoid data leakage and ensure consistency in the scaling process. Then, Bayesian optimization was employed to identify the optimal values of the algorithms' hyperparameters, as specified in Table 2. To prevent overfitting, k-fold cross-validation error was used as the objective function of the optimization algorithm. As illustrated in Figure 9, the results of Bayesian optimization reveal that convergence was achieved for all three ML models, indicating that the optimization process successfully found hyperparameter values that maximize the performance of each algorithm. The consistent convergence of Bayesian optimization across iterations highlights its effectiveness in navigating complex hyperparameter spaces. This stability and reliability reinforce its suitability for fine-tuning ML models. Additionally, this convergence validates the optimization strategy and suggests that the chosen hyperparameters are likely to generalize well across various datasets, thereby enhancing the versatility and applicability of the optimized models.

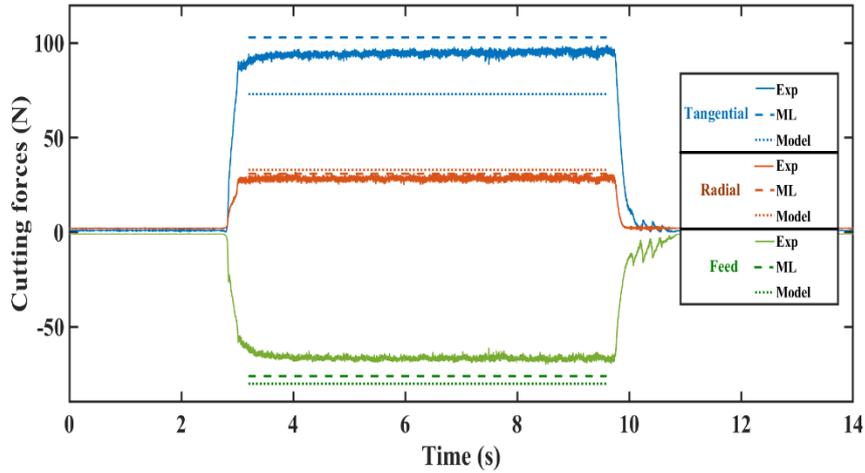


Figure 37. A sample of measured and predicted cutting forces (feed rate: 0.05 mm/rev; depth of cut: 0.5 mm; cutting speed: 100 m/min; nose radius: 0.2 mm; hone radius: 15  $\mu$ m).

Table 13. Comparison of cutting forces: Mechanistic model vs. experiment tests.

Machining parameters		Tool geometry			Experimental forces			Analytical model (decreasing pattern with zeta=1)			Analytical model (increasing-decreasing pattern with zeta=1)		
feed rate	depth	cutting speed	nose radius	hone radius	F <sub>x</sub>	F <sub>y</sub>	F <sub>z</sub>	F <sub>x</sub>	F <sub>y</sub>	F <sub>z</sub>	F <sub>x</sub>	F <sub>y</sub>	F <sub>z</sub>
(mm/rev)	(mm)	(m/min)	(mm)	( $\mu$ m)									
0.1	1	125	0.4	30	330	77	194	270	87	221	296	102	233
0.15	1.5	150	0.8	5	620	115	240	535	118	223	541	129	244
0.05	1.5	150	0.2	15	300	37	170	205	22	167	224	30	228
0.05	1.5	150	0.8	5	280	70	160	184	52	97	209	63	118
0.05	1	125	0.4	30	195	59	140	157	71	181	183	102	263
0.15	0.5	100	0.2	15	220	50	90	181	38	94	188	46	115
0.15	1.5	100	0.8	5	700	145	310	527	114	212	534	125	233
0.1	1	125	0.8	5	350	115	150	238	84	105	242	95	118
0.1	1	125	0.2	15	345	60	170	250	31	150	263	38	191
0.05	0.5	100	0.2	15	95	29	65	67	22	55	73	30	76

0.15	1.5	100	0.2	15	690	82	345	545	38	282	565	46	344
0.05	1.5	100	0.4	30	310	65	215	234	71	272	273	105	377
0.15	0.5	100	0.8	5	300	115	155	186	113	113	188	124	124
0.1	1.5	125	0.4	30	510	125	300	405	89	331	444	122	404
0.15	1	125	0.4	30	470	145	210	385	105	261	411	139	344
0.15	0.5	150	0.2	15	230	67	95	184	39	97	191	47	118
0.15	1.5	150	0.2	15	620	70	280	554	39	291	573	47	352
0.05	0.5	150	0.2	15	85	22	57	68	22	56	74	31	76
0.1	1	100	0.4	30	370	130	200	268	89	221	294	122	304
0.1	1	150	0.4	30	340	107	150	273	90	223	299	123	306
0.15	1	100	0.4	30	475	110	263	382	105	260	408	138	343
0.05	1	100	0.4	30	205	55	140	156	70	182	182	104	265
0.05	1	150	0.4	30	203	53	135	159	70	182	185	103	264
0.1	1.5	100	0.4	30	510	90	300	402	89	330	441	111	455
0.15	1.5	100	0.4	30	705	110	345	573	105	389	612	136	473
0.05	1.5	150	0.4	30	300	57	210	239	99	273	278	103	396
0.1	1.5	150	0.4	30	500	92	285	410	90	334	449	109	408
0.1	0.5	125	0.4	30	170	65	95	135	87	111	148	102	122
0.05	0.5	150	0.8	5	125	55	70	72	52	54	74	60	63
0.05	0.5	100	0.8	5	140	70	85	71	50	51	73	62	62
0.15	1.5	150	0.4	30	680	105	320	584	106	396	622	139	402
0.15	0.5	150	0.8	5	260	80	120	178	119	119	181	130	130

---

**R<sup>2</sup> for increasing-decreasing pressure distribution:** Tangential force: 85%; Feed force: 71%; Radial force: 67%.

---

**R<sup>2</sup> for decreasing pressure distribution:** Tangential force: 77%; Feed force: 75%; Radial force: 68%.

---

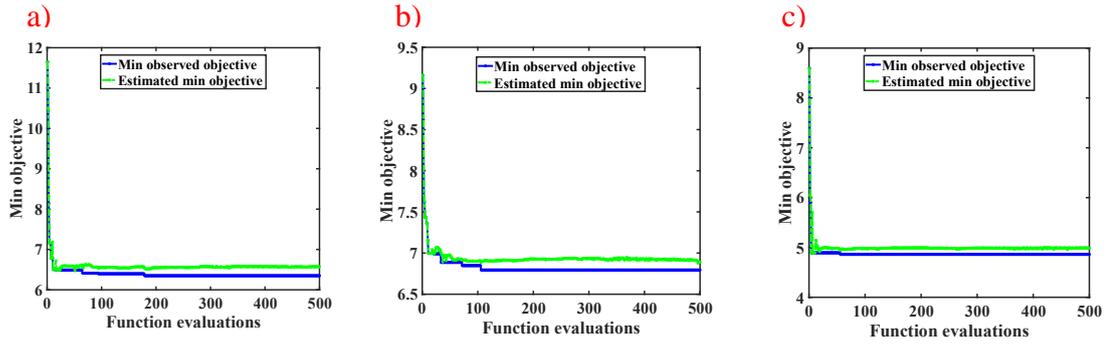


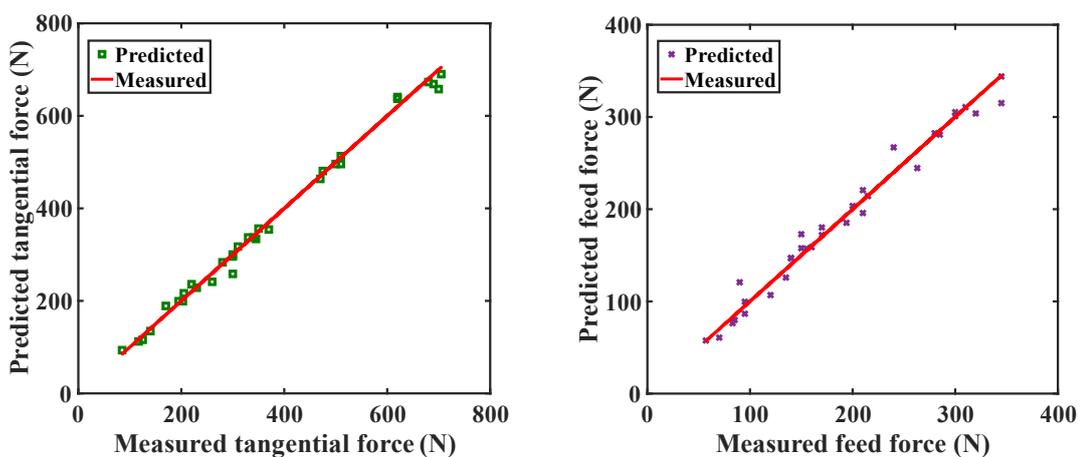
Figure 38. Progression of hyperparameter tuning in LSBoost model: a) Tangential force, b) Feed force and c) Radial force.

Table 14. Hyperparameter optimization results.

ML Models	Hyperparameters	Tangential force	Feed force	Radial force
SVR	Box Constraint	245.16	57.47	6.97
	Epsilon	8.86	11.85	6.94
	Kernel Function	Linear	3 <sup>rd</sup> order polynomial	Linear
Random Forest	Minimum leaf size	1	1	1
	No. predictors to sample	5	5	3
	No. of Trees	6	4	6
	In bag Fraction	0.95	0.91	0.74
LSBoost	No. of Learning Cycles	499	223	35
	Learning rate	0.06	0.04	0.34
	Minimum leaf size	1	5	2
	Maximum No. of splits	1	18	1
	No. of variables to sample	2	6	6

Followed by hyperparameter optimization, the regression analysis was performed with the PIML models. The obtained results are presented in Figure 10 and Tables 3 and 4. As demonstrated, all three machine learning models achieved remarkable accuracy in predicting cutting forces, which is evident from the close agreement between the predicted forces and the forces measured experimentally by the dynamometer. Among the ML models tested, LSBoost yielded the best performance, with SVR and RF showing

comparable performance for certain force components. It should be mentioned that the increasing-decreasing pressure distribution with zeta set to one slightly outperforms the decreasing pattern. The determination coefficient ( $R^2$ ) values of LSBoost algorithm for increasing-decreasing force model are 98%, 97% and 93% for tangential, feed and radial forces, with RMSE values of 14.93, 12.37 and 7.94, respectively. These values indicate the remarkable performance of the PIML model in predicting cutting forces in the turning process, considering both hone radius and nose radius effects. The determination coefficient measures the proportion of the variability in the cutting forces that can be explained by the model. A high R-squared value of more than 93% indicates that the model's features explain approximately 93% of the variability in cutting forces. Additionally, the low RMSE values signify a small average difference between predicted and actual cutting forces, suggesting close agreement between the model's predictions and the actual values. The obtained results demonstrate 13-25% improvement over the mechanistic turning model, highlighting the accuracy and effectiveness of the proposed approach in capturing the complex relationship between cutting forces, machining parameters and tool geometry. To further assess the accuracy of the PIML model, predictions were made for the unseen test dataset. The results indicated that the hybrid model effectively predicted cutting forces for this dataset as well, achieving  $R^2$  values of 97% for tangential force, 97% for feed force, and 91% for radial force. This validation on unseen test data confirms the robustness and reliability of the model across different datasets.



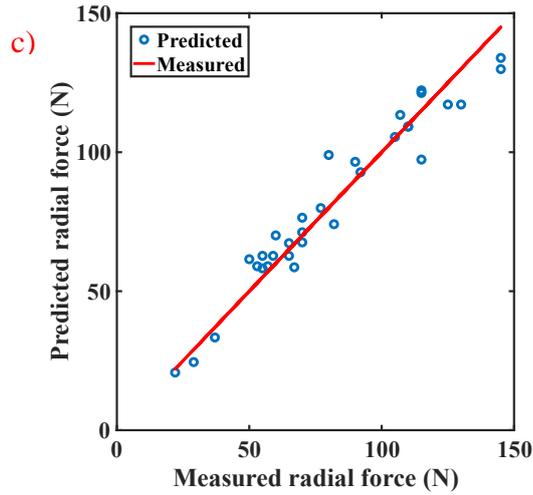


Figure 39. Regression curves of LSBoost model for: a) tangential; b) feed and c) radial forces.

Table 15. Performance metrics of the PIML algorithms for increasing-decreasing force model.

ML Models	Metrics	Tangential force	Radial force	Feed force
LSBoost	RMSE	14.93	7.94	12.37
	Adj. R <sup>2</sup>	98%	93%	97%
SVR	RMSE	16.88	14.85	14.88
	Adj. R <sup>2</sup>	98%	80%	96%
Random Forest	RMSE	23.19	10.22	20.81
	Adj. R <sup>2</sup>	98%	89%	94%

Table 16. Performance metrics of the PIML algorithms for decreasing force model.

ML Models	Metrics	Tangential force	Radial force	Feed force
LSBoost	RMSE	15.13	9.96	13.15
	Adj. R <sup>2</sup>	97%	90%	97%

SVR	RMSE	17.06	10.39	18.33
	Adj. R <sup>2</sup>	97%	89%	95%
Random Forest	RMSE	23.66	9.07	23.95
	Adj. R <sup>2</sup>	96%	92%	92%

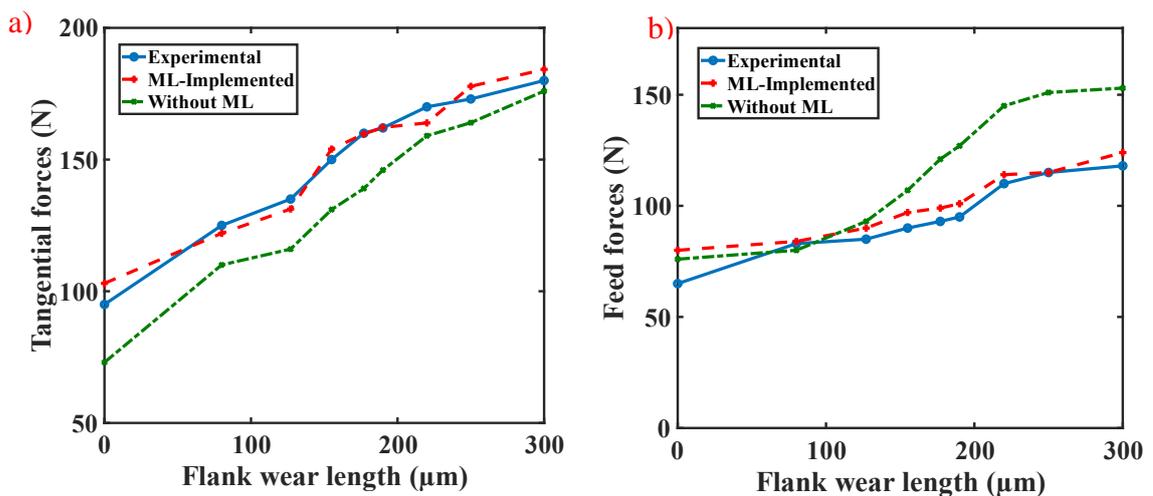
In summary, integrating the thermo-mechanical model into the ML algorithm not only enhanced the accuracy of the turning model by over 13-25%, but also this physics-informed model eliminated the need for numerous experimental tests. According to the obtained results, highly accurate predictions of cutting forces was achieved with a limited number of experimental tests (i.e., 32 tests). Such a hybrid physics-informed system bridges the gap between industry and research, where, the necessity for extensive datasets in conventional data-driven ML models poses a significant challenge for the application of these models in real machining environments, limiting their scope to research academy.

### 6.3.2. Flank wear effect on the cutting forces

Followed by the improving the accuracy of the mechanistic turning model through ML algorithms, the effect of flank wear on the cutting forces was investigated. For this purpose, the wear tests were carried out and the measured forces were compared with the model-estimated forces, with and without PIML implementation. In the developed PIML model, the wear length was also considered as one of the inputs of the ML algorithms, along with the machining parameters, hone radius, nose radius, and the cutting forces estimated by the wear-considered force model. Initially, the friction behavior on the worn surface was analyzed using an iterative numerical approach. The goal was to calibrate the friction coefficient in a way that the relative error between the measured cutting forces and the model-estimated forces was minimized. For this purpose, force modeling was executed with various values of the friction coefficient on the worn surface, and the one leading to the least difference between the experimental and analytical results was selected as the friction coefficient in the worn surface. According to the obtained results the friction coefficient in the worn surface varies between 0.7-0.9 for various cutting conditions. Also, various pressure distribution types and zeta values were examined to determine the most ideal conditions resulting in accurate predictions. Among these, the

increasing-decreasing pressure distribution with zeta equal to 1 resulted in the most accurate estimations. The comparison of the model-estimated and experimental forces is illustrated in Figure 40 and Figure 41, for two specific machining setups. For the entire dataset, the regression curves are shown in Figure 42, and the corresponding performance metrics (i.e.,  $R^2$  and RMSE values) are summarized in Table 17.

According to the obtained results, without employing the ML algorithms, the prediction accuracy of the model is 93% for the tangential force, 80% for the feed force, and 71% for the radial force. However, with the implementation of the PIML model, prediction accuracy increased to 97-98%, highlighting the superior performance of the proposed physics-informed ML model in accurately predicting turning forces in the presence of flank wear effects. This significant improvement demonstrates the effectiveness of integrating machine learning with analytical models, resulting in more reliable and precise predictions, even with limited number of experimental tests.



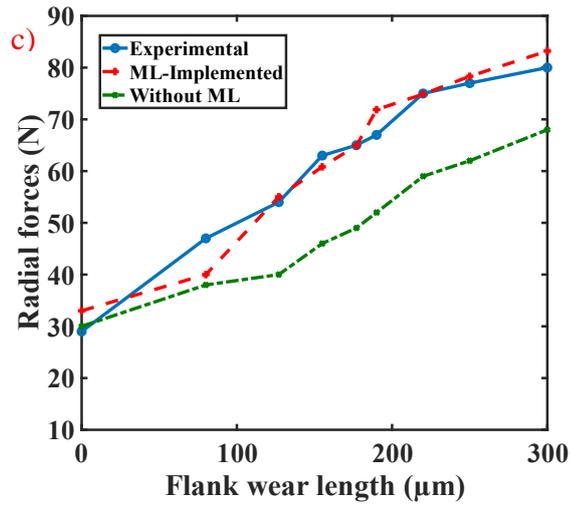
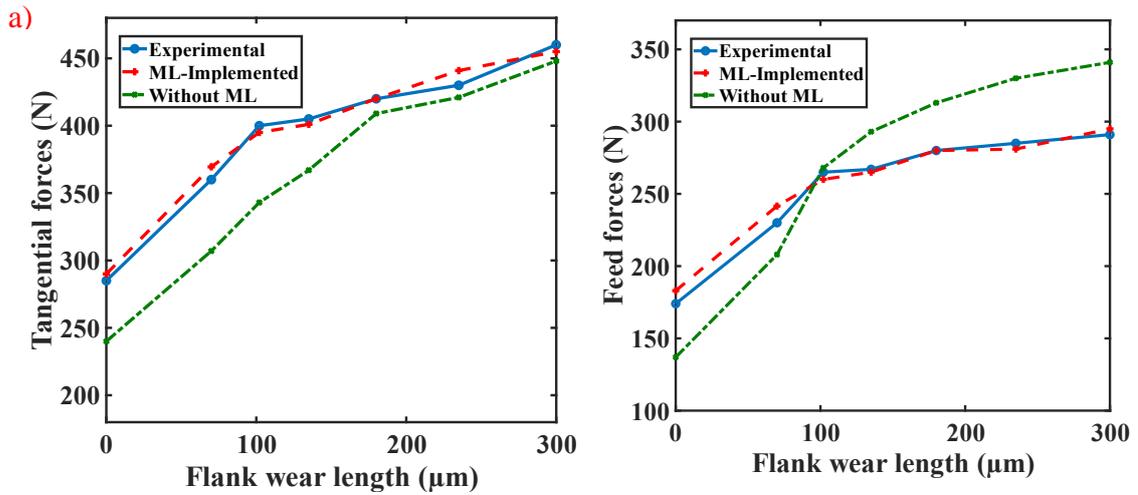


Figure 40. Model-estimated vs. experimental cutting forces: a) tangential; b) feed and c) radial forces. (feed rate= 0.05 mm/rev, depth of cut= 0.5 mm, cutting speed= 100 m/min, nose radius= 0.2 mm, hone radius= 15 μm)



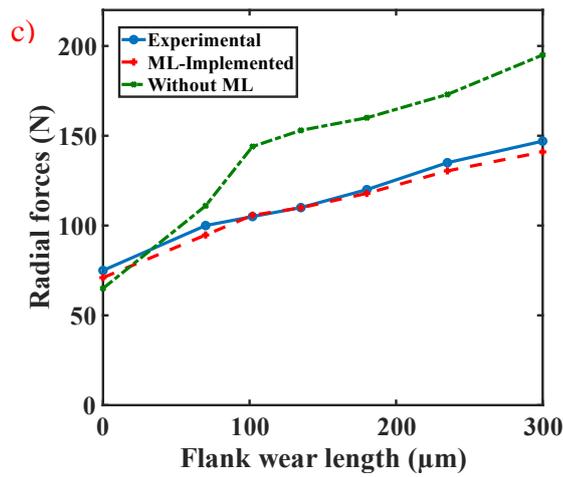
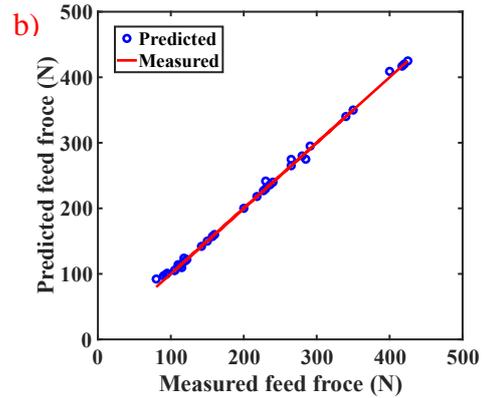
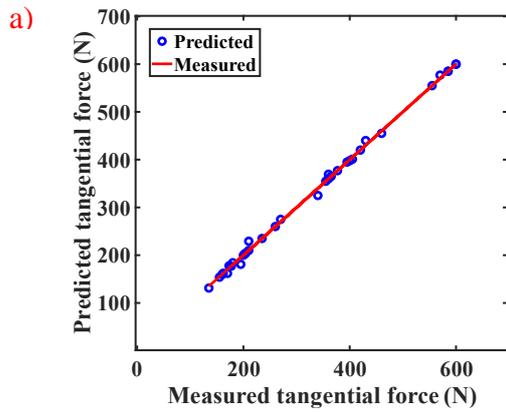


Figure 41. Model-estimated vs. experimental cutting forces: a) tangential; b) feed and c) radial forces. (feed rate= 0.05 mm/rev, depth of cut= 1.5 mm, cutting speed= 100 m/min, nose radius= 0.8 mm, hone radius= 5 μm)



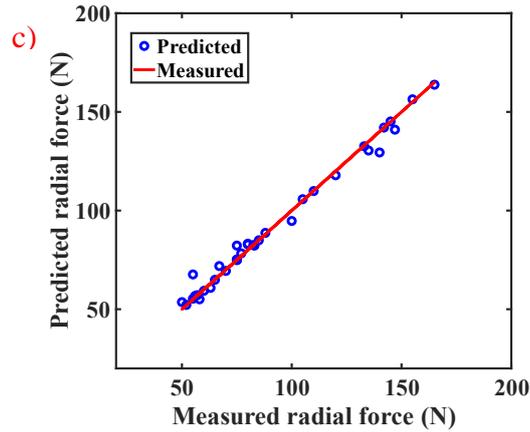


Figure 42. Regression curves of LSBoost algorithm for wear-considered force model:  
a) tangential; b) feed and c) radial forces  
(wear length varies between 45  $\mu\text{m}$  and 300  $\mu\text{m}$ ).

Table 17. Performance metrics of wear-included force model: with and without ML implementation.

ML Models	Metrics	Tangential force	Radial force	Feed force
Without ML	RMSE	37.51	30.58	47.17
	Adj. $R^2$	93%	71%	80%
LSBoost	RMSE	4.98	4.22	5.64
	Adj. $R^2$	98%	97%	98%
SVR	RMSE	5.38	5.48	6.23
	Adj. $R^2$	98%	96%	97%
Random Forest	RMSE	16.47	8.32	10.85
	Adj. $R^2$	97%	94%	97%

To evaluate the model's generalizability, a crucial aspect is whether the model can accurately estimate cutting forces for unseen conditions or not. This was investigated by applying the PIML-based model to unseen dataset without calibrating and updating the friction coefficient. In this respect, modeling was conducted using the average friction coefficient of 0.8 (obtained from the calibrated model), and the estimated cutting forces were compared with experimental results. The ML prediction applied to the unseen dataset reveals  $R^2$  value of 97% for the tangential force, 96% for the feed force, and 93% for the radial force.

The remarkable predictive capability on unseen data indicates ML's ability to generalize well beyond the training dataset, offering robust insights into process behavior even in unfamiliar conditions. This finding highlights the importance of PIML implementation in predictive modeling for enhanced accuracy and reliability, particularly in complex scenarios involving tool wear.

### **6.3.3. Reverse ML model for wear length prediction**

In this section, a reverse ML model was developed to predict the wear length based on cutting forces, machining parameters, hone radius, and nose radius. Since the accuracy of the wear-considered force model was improved through the physics-informed ML model, this enhanced model was used to generate a reliable training dataset for wear prediction, avoiding numerous expensive and time-consuming experimental wear tests. As explained in Section 6.3.2, the PIML-enhanced force model can predict cutting forces in the presence of flank wear with high accuracy ( $R^2$  exceeding 97%). Therefore, this model was employed to produce the training dataset for the ML algorithm to predict wear length. In this reverse ML model, wear length is the output, while the inputs are the improved cutting forces, machining parameters, nose radius and hone radius.

Initially, hyperparameter optimization was carried out using Bayesian optimization, followed by regression analysis using three ML algorithms: SVR, RF and LSBoost. The regression curves are illustrated in Figure 43, and the corresponding performance metrics are summarized in Table 18. As demonstrated, all ML models accurately predict the wear length, with LSBoost slightly showing better performance. For the entire dataset, the  $R^2$  values of the LSBoost model is 97%, and for the unseen test dataset, it is 94%. It should be mentioned for the unseen dataset, experimental tests were performed to validate the predicted wear length. These high  $R^2$  values along with small RMSE values indicate the superior performance of the proposed PIML model in accurately predicting the wear length according to the machining parameters and cutting forces. As previously demonstrated, incorporating the mechanistic force model into the ML algorithms eliminated the need for numerous experimental tests to train the ML model, while still achieving highly accurate predictions.

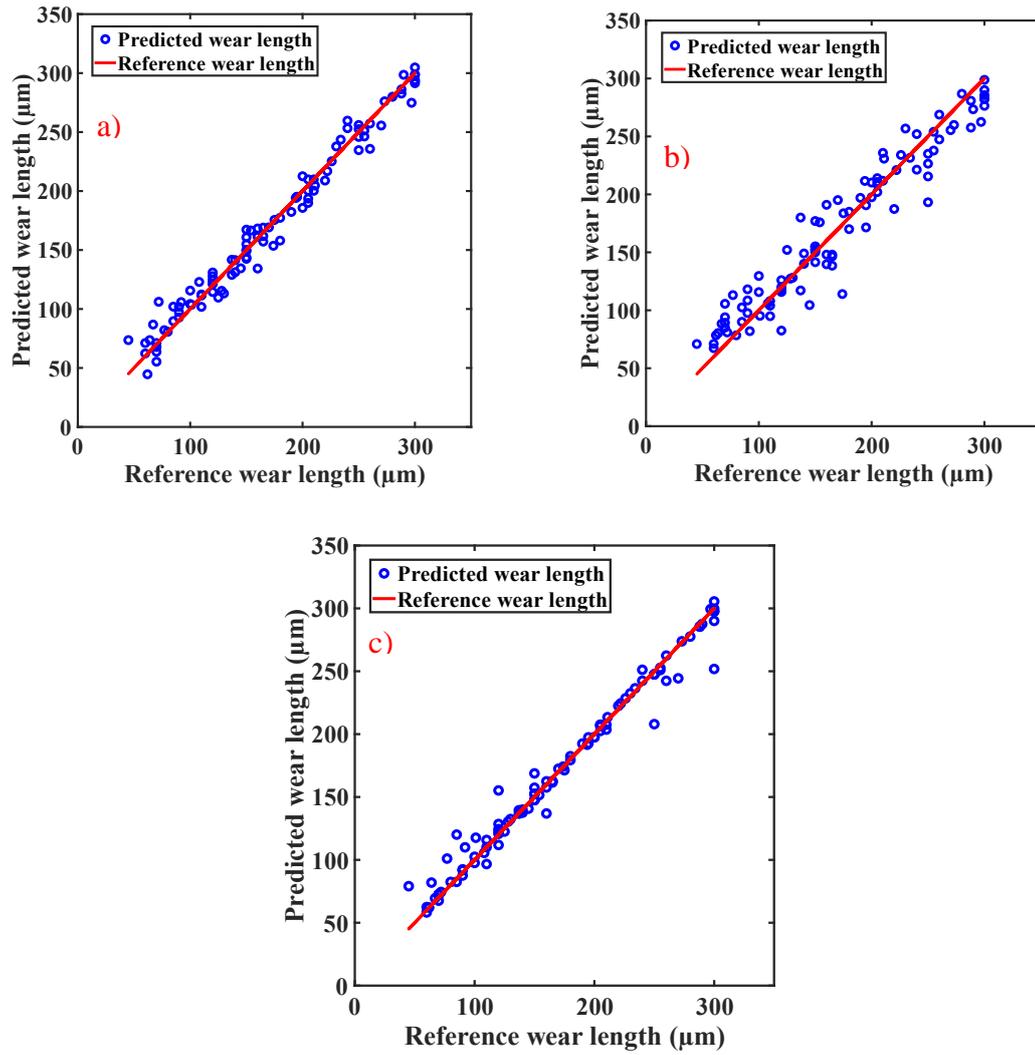


Figure 43. Regression curves of employed ML models for wear prediction: a) LSBoost; b) RF and c) SVR. (feed rate= [0.05, 0.1, 0.15, 0.2] mm/rev; depth of cut= [0.5, 1, 1.5, 2] mm; cutting speed= [100, 125, 150] m/min; nose radius= [0.2, 0.4, 0.8] mm; hone radius= [5, 15, 30]  $\mu\text{m}$ ).

Table 18. Performance metrics of ML models for wear prediction.

<b>Performance metrics</b>	<b>LSBoost</b>	<b>SVR</b>	<b>RF</b>
RMSE	10.88	11.06	19.54
Adj. R2	97%	97%	93%

The results of this study demonstrate that the proposed model accurately accounts for the influence of flank wear on cutting forces while also predicting wear length based on machining parameters, nose radius, hone radius, and cutting forces. Notably, this is achieved without the need for numerous time-consuming and expensive experimental tests, while still achieving high accuracy. Predicting tool wear in machining processes is essential for sustaining production efficiency and quality. It allows for preventative replacement before catastrophic failure, optimizing tool life for maximum efficiency without sacrificing part quality. As a result, downtime and costs are reduced due to tool changes and scrapped parts, and surface finish and dimensional accuracy enhance throughout the machining process. With such a critical advancement in the field of machining process monitoring, tool wear prediction through ML-improved cutting force model has significant implications for productivity, quality control, and competitiveness in the manufacturing domain.

#### **6.4. Summary**

This study proposed a novel physics-informed machine learning (PIML) model to predict wear length based on cutting forces, machining parameters, and tool geometry. This hybrid model addresses the limitations of conventional data-driven ML models by reducing the need for extensive and expensive wear tests, while simultaneously enhancing the accuracy of the analytical wear-included force model. To achieve this, a thermo-mechanical force model was established to calculate the cutting forces in the turning process, considering the effects of flank wear, nose radius, and hone radius (i.e., edge forces). First, the accuracy of this model was improved through the PIML, and subsequently it was used to generate a training dataset for another complementary reverse ML model to predict tool wear length, thereby streamlining the tool wear prediction process and eliminating the resource-intensive task of conducting numerous experimental tests. The following results were drawn from the present study:

- Without considering wear effects, the thermo-mechanical model could estimate the cutting forces with a prediction accuracy of 85% for tangential force, 71% for feed force, and 67% for radial force. However, integrating the PIML model significantly enhanced the prediction accuracy by 13% to 25%.

- The wear-included force model, which utilized calibrated friction coefficient for the worn surface, achieved a prediction accuracy of 93% for the tangential force, 80% for the feed force, and 71% for the radial force. By incorporating the PIML model, prediction accuracy increased to 97-98% for the entire training dataset, and 93-97% for the unseen test dataset.
- Leveraging the high prediction accuracy achieved by the PIML-enhanced force model, a training dataset was generated for wear length prediction. This enabled the development of a highly accurate reverse ML model that predicts wear length based on cutting forces, tool geometry, and machining parameters.
- Among the various ML algorithms employed, LSBoost demonstrated superior performance in terms of  $R^2$  and RMSE values, while SVR and RF showed comparable performance for certain force components.
- The proposed physics-informed ML model accurately predicts tool wear length by incorporating machining parameters, tool geometry, and cutting forces, achieving high prediction accuracy ( $R^2 > 97\%$ ) without extensive wear tests. This capability enables precise tool wear monitoring, enhancing production efficiency and making the model applicable to various industrial machining processes.

## 7. CONCLUSION

### 7.1. Conclusion

This thesis presents a comprehensive investigation into the development and application of advanced machine learning (ML) models integrated with physics-based simulations to enhance the accuracy and efficiency of milling and turning process monitoring. The overarching aim was to improve prediction accuracy for various machining parameters and tool wear, thereby optimizing manufacturing processes and reducing the need for extensive experimental testing. The key findings and contributions from each chapter are summarized below, highlighting their implications for the industry and future research directions.

In the first study, a hybrid model combining physics-based simulations and ML algorithms was developed to predict cutting forces during the milling process. By employing support vector regression (SVR), random forest (RF), and least-squares boosting (LSBoost), the model achieved high prediction accuracy, significantly improving upon traditional mechanistic models. The hybrid approach demonstrated remarkable accuracy, even for unseen datasets and diverse materials, including Steel 1050, Aluminum 7075-T6, Ti6Al4V, and Inconel 625. The integration of mechanistic milling models into the ML framework allowed the discovery of complex relationships between cutting forces, machining parameters, cutting conditions, and thermo-mechanical properties. This method not only reduced the number of required experimental tests but also enabled the continuous enhancement of the ML database through the application of IoT and Industry 4.0 principles. The proposed physics-based approach proved to be highly practical for real-world applications, providing a robust foundation for further advancements in simulation accuracy and process optimization.

The second study focused on developing a process simulation-based ML algorithm for monitoring and detecting tool-related faults in milling processes. By training the ML models solely on simulation data, the need for costly and time-consuming laboratory tests was eliminated. The random forest algorithm emerged as the most effective, achieving a

94% accuracy rate in predicting tool-related faults. This study highlighted the potential for significant improvements in fault monitoring systems, suggesting that future research could enhance performance by integrating additional data sources and adopting more advanced ML algorithms. The approach demonstrated a transformative potential in tool condition monitoring by streamlining the detection process, reducing costs, and improving manufacturing efficiency. These findings, supported by actual measurement data, underscore the method's viability for industrial applications and its ability to foster ongoing research and development efforts in tool condition monitoring.

The third study aimed to develop an intelligent real-time fault detection system for unmanned manufacturing using a novel hybrid ML approach. By utilizing highly accurate simulation results from physics-based models instead of experimental data, the system achieved over 96% accuracy in predicting machining parameters. The method demonstrated remarkable accuracy and reliability in real-time applications, with statistical error analysis indicating a notably confined error distribution. The system was validated through the machining of a complex free-form workpiece, proving its applicability in various industrial contexts. The proposed approach is particularly suitable for deployment in unmanned manufacturing applications, serving purposes such as process monitoring, fault detection, and parameter optimization. Additionally, the method's adaptability for industrial applications, such as utilizing CNC controller current signals instead of traditional sensors, further highlights its practicality and potential for wide-scale implementation.

The final study introduced a physics-informed machine learning (PIML) model for predicting tool wear length in turning processes. By integrating a thermo-mechanical force model with advanced ML algorithms, the study addressed the limitations of conventional data-driven models, reducing the dependency on extensive wear tests while enhancing prediction accuracy. The PIML model achieved high accuracy in predicting cutting forces and tool wear, demonstrating its capability to generalize across different materials and machining conditions. This hybrid model leveraged a comprehensive understanding of machining dynamics and tool wear mechanisms, achieving prediction accuracies of over 97% for the training dataset and 93-97% for the unseen test dataset. The study's findings highlight the model's potential to revolutionize tool wear prediction, offering a reliable, efficient, and scalable solution for modern manufacturing processes.

In conclusion, this thesis has significantly advanced the field of machining process monitoring by developing and validating innovative hybrid models that combine physics-based simulations with machine learning algorithms. These models not only improve prediction accuracy for cutting forces and tool wear but also reduce the need for extensive experimental testing, making them highly practical for industrial applications. The findings underscore the potential of integrating IoT and Industry 4.0 principles to continuously enhance the ML database and further improve monitoring systems. Future research should focus on expanding the scope of these models to include additional fault types and machining processes, integrating more advanced ML techniques, and exploring their deployment in diverse industrial settings. By addressing these future directions, the manufacturing industry can achieve greater efficiency, cost reduction, and quality improvement, fully realizing the benefits of these advanced monitoring systems.

The contributions of this research have laid a strong foundation for the continued development and application of intelligent monitoring systems in manufacturing. By leveraging the strengths of both physics-based simulations and machine learning, these systems can provide real-time, accurate, and reliable insights into machining processes, thereby enhancing decision-making and operational efficiency. As the industry moves towards greater automation and the adoption of smart technologies, the methodologies developed in this thesis will play a crucial role in shaping the future of manufacturing, driving innovation, and maintaining a competitive edge in an increasingly digital and interconnected world.

## **7.2. Future Research Directions**

This thesis has laid a strong foundation for the development and application of advanced machine learning (ML) models integrated with physics-based simulations for milling and turning process monitoring. Building on this work, several promising areas for future research have been identified:

### **1. Calculating Cutting Forces from CNC Controller Data Using ML:**

Developing ML models to directly calculate cutting forces from data obtained from CNC controllers, leveraging real-time data for enhanced monitoring and control.

## **2. Process Optimization Using ML and Identified Process Parameters:**

Utilizing ML algorithms to optimize machining processes based on the parameters identified through simulation and experimental data, aiming to improve efficiency, quality, and cost-effectiveness.

## **3. Optimization of Special Milling Tools Using ML:**

Applying ML algorithms to optimize the performance of special milling tools, such as variable pitch, serrated tools, and crest cuts, to achieve better machining outcomes.

## **4. Predicting Orthogonal Data for Cutter-Workpiece Combinations:**

Developing ML models to predict orthogonal data for various cutter-workpiece combinations without conducting extensive orthogonal tests, thus saving time and resources.

## **5. Machine Tool Parameter Identification Using Cutting Data:**

Utilizing ML algorithms to identify specifications of machine tool components such as spindles and servo drives from cutting data, enhancing the understanding and control of machining operations.

## **6. Analytical Modeling of Surface Roughness and Enhancement with ML:**

Creating analytical models for surface roughness in milling operations and enhancing these models using ML algorithms to improve surface quality predictions and process optimization.

## **7. Tool Specification Identification Using ML:**

Employing ML techniques to identify and optimize tool specifications, leading to better tool performance and longer tool life.

These future research directions promise to further enhance the capabilities of intelligent monitoring systems in manufacturing, contributing to greater efficiency, accuracy, and adaptability in machining processes. By exploring these areas, the potential of machine learning in the context of advanced manufacturing can be fully realized, driving innovation and maintaining a competitive edge in the industry.

## REFERENCES

- [1] C. Brecher, M. Esser, S. Witt, Interaction of manufacturing process and machine tool, *CIRP Ann.* 58 (2009) 588–607.
- [2] K. Jemielniak, Commercial tool condition monitoring systems, *Int. J. Adv. Manuf. Technol.* 15 (1999) 711–721.
- [3] S.Y. Liang, D.A. Dornfeld, Tool wear detection using time series analysis of acoustic emission, (1989).
- [4] S. Dimla DE, The correlation of vibration signal features to cutting tool wear in a metal turning operation, *Int. J. Adv. Manuf. Technol.* 19 (2002) 705–713.
- [5] T.J. Ko, D.W. Cho, M.Y. Jung, On-line monitoring of tool breakage in face milling using a self-organized neural network, *J. Manuf. Syst.* 14 (1995) 80–90.
- [6] S.C. Lin, R.J. Lin, Tool wear monitoring in face milling using force signals, *Wear.* 198 (1996) 136–142.
- [7] A. Kothuru, S.P. Nooka, R. Liu, Cutting Process Monitoring System Using Audible Sound Signals and Machine Learning Techniques: An Application to End Milling, in: *Int. Manuf. Sci. Eng. Conf.*, American Society of Mechanical Engineers, 2017: p. V003T04A050.
- [8] B. Sick, On-line and indirect tool wear monitoring in turning with artificial neural networks: a review of more than a decade of research, *Mech. Syst. Signal Process.* 16 (2002) 487–546.
- [9] R. Teti, K. Jemielniak, G. O'Donnell, D. Dornfeld, Advanced monitoring of machining operations, *CIRP Ann.* 59 (2010) 717–739.
- [10] M. Cheng, L. Jiao, P. Yan, H. Jiang, R. Wang, T. Qiu, X. Wang, Intelligent tool wear monitoring and multi-step prediction based on deep learning model, *J. Manuf. Syst.* 62 (2022) 286–300.
- [11] Y. Altintas, A.A. Ber, Manufacturing automation: metal cutting mechanics, machine tool vibrations, and CNC design, *Appl. Mech. Rev.* 54 (2001) B84–B84.
- [12] W.A. Kline, R.E. DeVor, J.R. Lindberg, The prediction of cutting forces in end milling with application to cornering cuts, *Int. J. Mach. Tool Des. Res.* 22 (1982) 7–22.
- [13] J.W. Sutherland, R. Devor, An improved method for cutting force and surface error prediction in flexible end milling systems, (1986).
- [14] J.W. Sutherland, A dynamic model of the cutting force system in the end milling

process, University of Illinois at Urbana-Champaign, 1987.

- [15] M. Materotti, An analysis of the milling process, *Trans. ASME*. 67 (1945) 233.
- [16] M.E. Merchant, *Basic mechanics of the metal-cutting process*, (1944).
- [17] M.E. Martellotti, An analysis of the milling process, *Trans. Am. Soc. Mech. Eng.* 63 (1941) 677–695.
- [18] F. Koenigsberger, A.J.P. Sabberwal, An investigation into the cutting force pulsations during milling operations, *Int. J. Mach. Tool Des. Res.* 1 (1961) 15–33.
- [19] E. Budak, Flexible Milling Force Model for Improved Surface Error Predictions, *Proc. 1992 Eng. Syst. Des. Anal.* 47 (1992) 89–94.
- [20] S. Smith, J. Tlustý, An overview of modeling and simulation of the milling process, (1991).
- [21] E. Budak, Y. Altintas, Peripheral milling conditions for improved dimensional accuracy, *Int. J. Mach. Tools Manuf.* 34 (1994) 907–918.
- [22] W.A. Kline, R.E. DeVor, The effect of runout on cutting geometry and forces in end milling, *Int. J. Mach. Tool Des. Res.* 23 (1983) 123–140.
- [23] K.M.Y. Law, A. Geddam, V.A. Ostafiev, A process-design approach to error compensation in the end milling of pockets, *J. Mater. Process. Technol.* 89 (1999) 238–244.
- [24] E.J.A. Armarego, N.P. Deshpande, Computerized end-milling force predictions with cutting models allowing for eccentricity and cutter deflections, *CIRP Ann.* 40 (1991) 25–29.
- [25] X.P. Li, A.Y.C. Nee, Y.S. Wong, H.Q. Zheng, Theoretical modelling and simulation of milling forces, *J. Mater. Process. Technol.* 89 (1999) 266–272.
- [26] E. Budak, Analytical models for high performance milling. Part I: Cutting forces, structural deformations and tolerance integrity, *Int. J. Mach. Tools Manuf.* 46 (2006) 1478–1488.
- [27] S. Vaishnav, A. Agarwal, K.A. Desai, Machine learning-based instantaneous cutting force model for end milling operation, *J. Intell. Manuf.* 31 (2020) 1353–1366.
- [28] T. Radhakrishnan, U. Nandan, Milling force prediction using regression and neural networks, *J. Intell. Manuf.* 16 (2005) 93.
- [29] P. Charalampous, Prediction of cutting forces in milling using machine learning algorithms and finite element analysis, *J. Mater. Eng. Perform.* 30 (2021) 2002–2013.
- [30] U. Zuperl, F. Cus, B. Mursec, T. Ploj, A generalized neural network model of ball-

- end milling force system, *J. Mater. Process. Technol.* 175 (2006) 98–108.
- [31] S. Al-Zubaidi, J.A. Ghani, C.H.C. Haron, Application of ANN in milling process: a review, *Model. Simul. Eng.* 2011 (2011) 1–7.
- [32] A. Gouarir, G. Martínez-Arellano, G. Terrazas, P. Benardos, S. Ratchev, In-process tool wear prediction system based on machine learning techniques and force analysis, *Procedia CIRP.* 77 (2018) 501–504.
- [33] B. Peng, T. Bergs, D. Schraknepper, F. Klocke, B. Döbbeler, A hybrid approach using machine learning to predict the cutting forces under consideration of the tool wear, *Procedia Cirp.* 82 (2019) 302–307.
- [34] V. Tandon, H. El-Mounayri, A novel artificial neural networks force model for end milling, *Int. J. Adv. Manuf. Technol.* 18 (2001) 693–700.
- [35] C.-H. Hsieh, J.-H. Chou, Y.-J. Wu, Optimal predicted fuzzy controller of a constant turning force system with fixed metal removal rate, *J. Mater. Process. Technol.* 123 (2002) 22–30.
- [36] V. Tandon, H. El-Mounayri, H. Kishawy, NC end milling optimization using evolutionary computation, *Int. J. Mach. Tools Manuf.* 42 (2002) 595–605.
- [37] J.F. Briceno, H. El-Mounayri, S. Mukhopadhyay, Selecting an artificial neural network for efficient modeling and accurate simulation of the milling process, *Int. J. Mach. Tools Manuf.* 42 (2002) 663–674.
- [38] H.K. Dave, H.K. Raval, Modelling of cutting forces as a function of cutting parameters in milling process using regression analysis and artificial neural network, *Int. J. Mach. Mach. Mater.* 8 (2010) 198–208.
- [39] K. Xu, Y. Li, J. Zhang, G. Chen, ForceNet: An offline cutting force prediction model based on neuro-physical learning approach, *J. Manuf. Syst.* 61 (2021) 1–15.
- [40] J. Xie, P. Hu, J. Chen, W. Han, R. Wang, Deep learning-based instantaneous cutting force modeling of three-axis CNC milling, *Int. J. Mech. Sci.* 246 (2023) 108153.
- [41] F. Kara, K. Aslantas, A. Çiçek, ANN and multiple regression method-based modelling of cutting forces in orthogonal machining of AISI 316L stainless steel, *Neural Comput. Appl.* 26 (2015) 237–250.
- [42] J. Wang, B. Zou, M. Liu, Y. Li, H. Ding, K. Xue, Milling force prediction model based on transfer learning and neural network, *J. Intell. Manuf.* 32 (2021) 947–956.
- [43] Y. Altintas, D. Aslan, Integration of virtual and on-line machining process control and monitoring, *CIRP Ann.* 66 (2017) 349–352.
- [44] R. Teti, D. Mourtzis, D.M. D’Addona, A. Caggiano, Process monitoring of

- machining, *CIRP Ann.* 71 (2022) 529–552.
- [45] E. Özlü, A. Ebrahimi Araghizad, E. Budak, Broaching tool design through force modelling and process simulation, *CIRP Ann.* 69 (2020) 53–56. <https://doi.org/10.1016/j.cirp.2020.04.035>.
- [46] G. Ambrogio, L. Filice, F. Longo, A. Padovano, Workforce and supply chain disruption as a digital and technological innovation opportunity for resilient manufacturing systems in the COVID-19 pandemic, *Comput. Ind. Eng.* 169 (2022) 108158.
- [47] A.E. Araghizad, F. Tehranizadeh, K. Kilic, E. Budak, Smart Tool-Related Faults Monitoring System Using Process Simulation-Based Machine Learning Algorithms, *J. Mach. Eng.* 23 (2023).
- [48] G. Byrne, D. Dornfeld, I. Inasaki, G. Ketteler, W. König, R. Teti, Tool condition monitoring (TCM)—the status of research and industrial application, *CIRP Ann.* 44 (1995) 541–567.
- [49] B. Denkena, R. Fischer, D. Euhus, T. Neff, Simulation based process monitoring for single item production without machine external sensors, *Procedia Technol.* 15 (2014) 341–348.
- [50] A.E. Araghizad, F. Pashmforoush, F. Tehranizadeh, K. Kilic, E. Budak, Improving milling force predictions: A hybrid approach integrating physics-based simulation and machine learning for remarkable accuracy across diverse unseen materials and tool types, *J. Manuf. Process.* 114 (2024) 92–107.
- [51] E. Budak, Y. Altintas, E.J.A. Armarego, Prediction of milling force coefficients from orthogonal cutting data, (1996).
- [52] M. Balazinski, E. Czogala, K. Jemielniak, J. Leski, Tool condition monitoring using artificial intelligence methods, *Eng. Appl. Artif. Intell.* 15 (2002) 73–80.
- [53] M.A. Elbestawi, T.A. Papazafiriou, R.X. Du, In-process monitoring of tool wear in milling using cutting force signature, *Int. J. Mach. Tools Manuf.* 31 (1991) 55–73.
- [54] I.N. Tansel, T.T. Arkan, W.Y. Bao, N. Mahendrakar, B. Shisler, D. Smith, M. McCool, Tool wear estimation in micro-machining.: Part I: tool usage–cutting force relationship, *Int. J. Mach. Tools Manuf.* 40 (2000) 599–608.
- [55] I.N. Tansel, T.T. Arkan, W.Y. Bao, N. Mahendrakar, B. Shisler, D. Smith, M. McCool, Tool wear estimation in micro-machining.: Part II: neural-network-based periodic inspector for non-metals, *Int. J. Mach. Tools Manuf.* 40 (2000) 609–620.
- [56] H. Saglam, A. Unuvar, Tool condition monitoring in milling based on cutting forces by a neural network, *Int. J. Prod. Res.* 41 (2003) 1519–1532.
- [57] E. Kuljanic, M. Sortino, TWEM, a method based on cutting forces—monitoring

- tool wear in face milling, *Int. J. Mach. Tools Manuf.* 45 (2005) 29–34.
- [58] H.Z. Li, H. Zeng, X.Q. Chen, An experimental study of tool wear and cutting force variation in the end milling of Inconel 718 with coated carbide inserts, *J. Mater. Process. Technol.* 180 (2006) 296–304.
- [59] M. Wang, J. Wang, CHMM for tool condition monitoring and remaining useful life prediction, *Int. J. Adv. Manuf. Technol.* 59 (2012) 463–471.
- [60] 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.
- [61] A.I. Azmi, Monitoring of tool wear using measured machining forces and neuro-fuzzy modelling approaches during machining of GFRP composites, *Adv. Eng. Softw.* 82 (2015) 53–64.
- [62] R.H.L. da Silva, M.B. da Silva, A. Hassui, A probabilistic neural network applied in monitoring tool wear in the end milling operation via acoustic emission and cutting power signals, *Mach. Sci. Technol.* 20 (2016) 386–405.
- [63] 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 (2015) 213–223.
- [64] G.S. Hong, M. Rahman, Q. Zhou, Using neural network for tool condition monitoring based on wavelet decomposition, *Int. J. Mach. Tools Manuf.* 36 (1996) 551–566.
- [65] S. Shankar, T. Mohanraj, A. Pramanik, Tool condition monitoring while using vegetable based cutting fluids during milling of inconel 625, *J. Adv. Manuf. Syst.* 18 (2019) 563–581.
- [66] X. Li, H.-X. Li, X.-P. Guan, R. Du, Fuzzy estimation of feed-cutting force from current measurement—a case study on intelligent tool wear condition monitoring, *IEEE Trans. Syst. Man, Cybern. Part C (Applications Rev.)* 34 (2004) 506–512.
- [67] S. Shankar, T. Mohanraj, Tool condition monitoring in milling using sensor fusion technique, in: *Proc. Malaysian Int. Tribol. Conf.*, 2015: pp. 322–323.
- [68] B. Kaya, C. Oysu, H.M. Ertunc, H. Ocak, A support vector machine-based online tool condition monitoring for milling using sensor fusion and a genetic algorithm, *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.* 226 (2012) 1808–1818.
- [69] M. Elangovan, K.I. Ramachandran, V. Sugumaran, Studies on Bayes classifier for condition monitoring of single point carbide tipped tool based on statistical and histogram features, *Expert Syst. Appl.* 37 (2010) 2059–2065.
- [70] 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.

- [71] J. Yang, J. Duan, T. Li, C. Hu, J. Liang, T. Shi, Tool wear monitoring in milling based on fine-grained image classification of machined surface images, *Sensors*. 22 (2022) 8416.
- [72] R. Daicu, G. Oancea, Methodology for measuring the cutting inserts wear, *Symmetry (Basel)*. 14 (2022) 469.
- [73] M. Kuntoğlu, A. Aslan, D.Y. Pimenov, Ü.A. Usca, E. Salur, M.K. Gupta, T. Mikolajczyk, K. Giasin, W. Kapłonek, S. Sharma, A review of indirect tool condition monitoring systems and decision-making methods in turning: Critical analysis and trends, *Sensors*. 21 (2020) 108.
- [74] X. Chuangwen, D. Jianming, C. Yuzhen, L. Huaiyuan, S. Zhicheng, X. Jing, The relationships between cutting parameters, tool wear, cutting force and vibration, *Adv. Mech. Eng.* 10 (2018) 1687814017750434.
- [75] L. Dan, J. Mathew, Tool wear and failure monitoring techniques for turning—a review, *Int. J. Mach. Tools Manuf.* 30 (1990) 579–598.
- [76] K.N. Shi, D.H. Zhang, N. Liu, S.B. Wang, J.X. Ren, S.L. Wang, A novel energy consumption model for milling process considering tool wear progression, *J. Clean. Prod.* 184 (2018) 152–159.
- [77] F.J. Alonso, D.R. Salgado, Analysis of the structure of vibration signals for tool wear detection, *Mech. Syst. Signal Process.* 22 (2008) 735–748.
- [78] Z. Huang, J. Zhu, J. Lei, X. Li, F. Tian, Tool wear monitoring with vibration signals based on short-time fourier transform and deep convolutional neural network in milling, *Math. Probl. Eng.* 2021 (2021) 1–14.
- [79] J.-H. Zhou, C.K. Pang, Z.-W. Zhong, F.L. Lewis, Tool wear monitoring using acoustic emissions by dominant-feature identification, *IEEE Trans. Instrum. Meas.* 60 (2010) 547–559.
- [80] X. Li, A brief review: acoustic emission method for tool wear monitoring during turning, *Int. J. Mach. Tools Manuf.* 42 (2002) 157–165.
- [81] R. Teti, G.F. Micheletti, Tool wear monitoring through acoustic emission, *CIRP Ann.* 38 (1989) 99–102.
- [82] A.E. Araghizad, F. Tehranizadeh, F. Pashmforoush, E. Budak, Milling process monitoring based on intelligent real-time parameter identification for unmanned manufacturing, *CIRP Ann.* (2024).
- [83] K. Wang, A. Wang, L. Wu, G. Xie, Machine Tool Wear Prediction Technology Based on Multi-Sensor Information Fusion, *Sensors*. 24 (2024) 2652.
- [84] K. Hosseinkhani, E. Ng, A combined empirical and numerical approach for tool

- wear prediction in machining, *Procedia CIRP*. 31 (2015) 304–309.
- [85] A. Ruiz, D. Guinea, L.J. Barrios, F. Betancourt, An empirical multi-sensor estimation of tool wear, *Mech. Syst. Signal Process.* 7 (1993) 105–119.
- [86] Z. Liu, C. Yue, X. Li, X. Liu, S.Y. Liang, L. Wang, Research on tool wear based on 3D FEM simulation for milling process, *J. Manuf. Mater. Process.* 4 (2020) 121.
- [87] F. Ducobu, P.-J. Arrazola, E. Rivière-Lorphèvre, E. Filippi, Finite element prediction of the tool wear influence in Ti6Al4V machining, *Procedia Cirp*. 31 (2015) 124–129.
- [88] Y.-C. Yen, J. Söhner, B. Lilly, T. Altan, Estimation of tool wear in orthogonal cutting using the finite element analysis, *J. Mater. Process. Technol.* 146 (2004) 82–91.
- [89] Y. Huang, S.Y. Liang, Modeling of cutting forces under hard turning conditions considering tool wear effect, *J. Manuf. Sci. Eng.* 127 (2005) 262–270.
- [90] E. Usui, T. Shirakashi, T. Kitagawa, Analytical prediction of cutting tool wear, *Wear*. 100 (1984) 129–151.
- [91] Y. Wang, M. Chang, X. Huang, Y. Li, J. Tang, Cutting tool wear prediction based on the multi-stage Wiener process, *Int. J. Adv. Manuf. Technol.* 129 (2023) 5319–5333.
- [92] Z. Li, R. Liu, D. Wu, Data-driven smart manufacturing: Tool wear monitoring with audio signals and machine learning, *J. Manuf. Process.* 48 (2019) 66–76.
- [93] B. Kilundu, P. Dehombreux, X. Chimentin, Tool wear monitoring by machine learning techniques and singular spectrum analysis, *Mech. Syst. Signal Process.* 25 (2011) 400–415.
- [94] X. Li, X. Qin, J. Wu, J. Yang, Z. Huang, Tool wear prediction based on convolutional bidirectional LSTM model with improved particle swarm optimization, *Int. J. Adv. Manuf. Technol.* 123 (2022) 4025–4039.
- [95] C. Cortes, V. Vapnik, Support-vector networks, *Mach. Learn.* 20 (1995) 273–297.
- [96] A.J. Smola, B. Schölkopf, A tutorial on support vector regression, *Stat. Comput.* 14 (2004) 199–222.
- [97] B. Scholkopf, A.J. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*, MIT press, 2018.
- [98] K.K. Ladha, The Condorcet jury theorem, free speech, and correlated votes, *Am. J. Pol. Sci.* (1992) 617–634.
- [99] M. Kearns, L. Valiant, Cryptographic limitations on learning boolean formulae and

- finite automata, *J. ACM.* 41 (1994) 67–95.
- [100] R.E. Schapire, The strength of weak learnability, *Mach. Learn.* 5 (1990) 197–227.
- [101] Y. Freund, R.E. Schapire, Experiments with a new boosting algorithm, in: *Icml*, Citeseer, 1996: pp. 148–156.
- [102] J.H. Friedman, Greedy function approximation: a gradient boosting machine, *Ann. Stat.* (2001) 1189–1232.
- [103] T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, in: *Proc. 22nd Acm Sigkdd Int. Conf. Knowl. Discov. Data Min.*, 2016: pp. 785–794.
- [104] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, Lightgbm: A highly efficient gradient boosting decision tree, *Adv. Neural Inf. Process. Syst.* 30 (2017).
- [105] J.N. Morgan, J.A. Sonquist, Problems in the analysis of survey data, and a proposal, *J. Am. Stat. Assoc.* 58 (1963) 415–434.
- [106] L. Breiman, J. Friedman, R. Olshen, C. Stone, *Cart, Classif. Regres. Trees.* (1984).
- [107] J.R. Quinlan, *C4. 5: programs for machine learning*, Elsevier, 2014.
- [108] J.R. Quinlan, Discovering rules from large collections of examples: a case study, *Expert Syst. Microelectron. Age.* (1979).
- [109] L. Breiman, Bagging predictors, *Mach. Learn.* 24 (1996) 123–140.
- [110] T.K. Ho, The random subspace method for constructing decision forests, *IEEE Trans. Pattern Anal. Mach. Intell.* 20 (1998) 832–844.
- [111] Y. Amit, D. Geman, Shape quantization and recognition with randomized trees, *Neural Comput.* 9 (1997) 1545–1588.
- [112] T.G. Dietterich, An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting and randomization, *Mach. Learn.* 32 (1998) 1–22.
- [113] L. Breiman, Random forests, *Mach. Learn.* 45 (2001) 5–32.
- [114] J.R. Quinlan, Induction of decision trees, *Mach. Learn.* 1 (1986) 81–106.
- [115] F. Pashmforoush, M. Seyedzavvar, A transfer learning-based machine learning approach to predict mechanical properties of different material types fabricated by selective laser melting process, *Proc. Inst. Mech. Eng. Part E J. Process Mech. Eng.* (2023) 09544089231215683.
- [116] Y. Chen, F. Li, S. Zhou, X. Zhang, S. Zhang, Q. Zhang, Y. Su, Bayesian optimization based random forest and extreme gradient boosting for the pavement density prediction in GPR detection, *Constr. Build. Mater.* 387 (2023) 131564.

- [117] R. Martinez-Cantin, Bayesian optimization with adaptive kernels for robot control, in: 2017 IEEE Int. Conf. Robot. Autom., IEEE, 2017: pp. 3350–3356.
- [118] F. Tehranizadeh, R. Koca, E. Budak, Investigating effects of serration geometry on milling forces and chatter stability for their optimal selection, *Int. J. Mach. Tools Manuf.* 144 (2019) 103425.
- [119] E. Özlü, A. Ebrahimi Araghizad, E. Budak, Broaching tool design through force modelling and process simulation, *CIRP Ann.* (2020). <https://doi.org/10.1016/j.cirp.2020.04.035>.
- [120] E. Budak, E. Ozlu, H. Bakioglu, Z. Barzegar, Thermo-mechanical modeling of the third deformation zone in machining for prediction of cutting forces, *CIRP Ann.* 65 (2016) 121–124.
- [121] F. Tehranizadeh, K.R. Berenji, E. Budak, Dynamics and chatter stability of crest-cut end mills, *Int. J. Mach. Tools Manuf.* 171 (2021) 103813.
- [122] R.L. Kegg, One-line machine and process diagnostics, *CIRP Ann.* 33 (1984) 469–473.
- [123] E.J.A. Armarego, A.J.R. Smith, Z.J. Gong, Four plane facet point drills—basic design and cutting model predictions, *CIRP Ann.* 39 (1990) 41–45.
- [124] T.K. Ho, Random decision forests, in: *Proc. 3rd Int. Conf. Doc. Anal. Recognit.*, IEEE, 1995: pp. 278–282.
- [125] E. Budak, Y. Altıntaş, E.J.A. Armarego, Prediction of milling force coefficients from orthogonal cutting data, *J. Manuf. Sci. Eng. Trans. ASME.* 118 (1996) 216–224. <https://doi.org/10.1115/1.2831014>.
- [126] F.C. Zegarra, J. Vargas-Machuca, A.M. Coronado, Tool wear and remaining useful life (RUL) prediction based on reduced feature set and Bayesian hyperparameter optimization, *Prod. Eng.* 16 (2022) 465–480.
- [127] E. Budak, L. Kops, Improving productivity and part quality in milling of titanium based impellers by chatter suppression and force control, *CIRP Ann.* 49 (2000) 31–36.
- [128] A. Moufki, A. Devillez, D. Dudzinski, A. Molinari, Thermomechanical modelling of oblique cutting and experimental validation, *Int. J. Mach. Tools Manuf.* 44 (2004) 971–989.
- [129] D. Ulutan, T. Özel, Determination of tool friction in presence of flank wear and stress distribution based validation using finite element simulations in machining of titanium and nickel based alloys, *J. Mater. Process. Technol.* 213 (2013) 2217–2237.
- [130] K. Aslantas, Ş. Ülker, Ö. Şahan, D.Y. Pimenov, K. Giasin, Mechanistic modeling of cutting forces in high-speed microturning of titanium alloy with consideration

of nose radius, *Int. J. Adv. Manuf. Technol.* 119 (2022) 2393–2408.

- [131] E. Budak, E. Ozlu, Development of a thermomechanical cutting process model for machining process simulations, *CIRP Ann.* 57 (2008) 97–100.